Knowledge-based systems as companions

Numan, John Han

Knowledge-Based Systems as Companions

Trust, Human Computer Interaction and Complex Systems

John Han Numan

Rijksuniversiteit Groningen

Knowledge-Based Systems as Companions

Trust, Human Computer Interaction and Complex Systems

Proefschrift

ter verkrijging van het doctoraat in de
Bedrijfskunde
aan de Rijksuniversiteit Groningen
op gezag van de
Rector Magnificus Dr. F. van der Woude
in het openbaar te verdedigen op
donderdag 14 mei 1998
des namiddags te 2.45 uur

door
John Han Numan
geboren op 26 januari 1962
te Enschede

Promotor
Prof. Dr. W. van Rossum

Co-promotor
Dr. R.J.J.M. Jorna

ISBN 90-72591-56-9

Promotiecommissie (dissertation committee)

Prof. Dr. Ir. N.J.I. Mars
Prof. Dr. T.F. Meijman
Prof. Dr. A.H. van der Zwaan

Uitgever: Labryrint Publication
Postbus 662
2900 AR Capelle a/d Ijssel
fax +31 (0) 10 2847382

Drukwerk: Ridderprint

ISBN 90-72591-56-9

Preface

The research described in this thesis was conducted at the faculty of Management and Organization (Faculteit Bedrijfskunde) of the University of Groningen, the Netherlands. The research project was conducted under the supervision of René Jorna and Wouter van Rossum. I would like to thank them for their enthusiasm, knowledge and, above all, support. They gave invaluable assistance in writing this thesis.

Writing this thesis required the co-operation of people in various disciplines. Their support was of great value to me. I would like to thank them here:

Many nurse schedulers, hearing therapists of the University Hospital Groningen (AZG) and hearing therapists of several hearing-aid shops in Groningen. I would like to thank them all for their time and patience.

The members of the dissertation committee, professor Mars, professor Meijman and professor van der Zwaan for their valuable comments on this thesis.

Jacques Boersma (IKS-produkten B.V. Groningen) for supplying the system ZKR to do several experiments and Bauke Lyklema for adjusting the system ZKR to meet the demands of my experiments.

The members of the workgroup DJOB (Ph.D. students in the faculty of Management and Organization) and the members of DISKUS. Especially my former colleagues Ad Breukel, Hans van den Broek, Sven van der Zee, Erik Huisman, Constantijn Heesen, Vincent Homburg, John Simons, Thomas Klos, Johan Oldenkamp, Dorin and Mirella Arion, Simon Sibum, Wout van Wezel, Margriet Offereins, Dieta Mietus, Derk Jan Kiewiet, Peter Muller, Fimke Zwerink, and Henk Gazendam.

Wim van Nieuwenhuijsen for his comments and suggestions on the statistical analysis of the data for this thesis.

My paranymphs Eric Barten and Harrie de Vries especially for walking in ‘ganzepas’ and for the party which is to come.

My parents, parents in law, sisters and brothers for always being there for me when I needed to have a break from it all.

Finally, I would like to thank Tineke and Stijn for sharing my involvement in this work and my life.

J. Han Numan
Groningen, March 1998

CONTENTS

1. INTRODUCTION 1

1.1 INTRODUCTION 1
1.1.1 Usability 1
1.1.2 Attitudes 2
1.1.3 Knowledge-based systems 4
1.1.4 Feedback 6
1.1.5 Magical thinking 7
1.1.6 Animism 8
1.1.7 Human-Computer co-operation 8
1.2 TRUST 9
1.2.1 CASH 1 10
1.2.2 CASH2 11
1.2.3 CASH 1 versus CASH 2 11
1.2.4 Conclusion 12
1.3 RESEARCH QUESTIONS 12
1.4 OVERVIEW 13
1.4.1 Research process 15
1.5 CONCLUSION 15

2. OVERVIEW OF HUMAN-COMPUTER INTERACTION 16

2.1 THEORETICAL BACKGROUND TO HUMAN-COMPUTER INTERACTION RESEARCH 16
2.1.1 Introduction 16
2.1.2 The ARK knowledge-based system 17
2.1.3 The task-oriented approach 18
2.1.3.1 Conclusion 19
2.1.4 The machine-oriented approach 19
2.1.4.1 Conclusion 20
2.1.5 The user-oriented approach 20
2.1.5.1 Conclusion 22
2.1.6 The (social) context-oriented approach 22
2.1.6.1 The impact a computer system has on the social context 22
2.1.6.2 The impact of the social context on the design of the Human-Computer Interaction component 24
2.1.7 Conclusion 25
2.2 PERSPECTIVE ON HCI 26
2.3 CONCLUSION 29

3. TRUST AND RELATED CONCEPTS 30

3.1 INTRODUCTION 30
3.2 RESEARCH ON TRUST 30
3.2.1 Functionality of trust 30
3.2.2 The content of trust 33
3.2.3 Conclusion 35
3.3 TRUST 36
3.3.1 Introduction 36
3.3.2 Definition of trust 36
3.3.3 Faith, trust and confidence 37
3.3.3.1 Introduction 37
3.3.3.2 Faith 38
3.3.3.3 Trust 38
3.3.3.4 Confidence 39
3.3.3.5 Conclusion 39
3.3.4 The dynamics of trust 39
3.4 MODEL OF TRUST 40
3.4.1 Trust 40
3.4.1.1 Trust I 41
3.4.1.2 Trust II 41

3.4.1.3 Trust III 42
3.4.2 Conclusion 42
3.5 RELATED CONCEPTS 43
3.5.1 Introduction 43
3.5.2 Complexity 43
3.5.3 Complicatedness 46
3.5.3.1 Conclusion 47
3.5.4 Uncertainty 47
3.5.4.1 Conclusion 48
3.5.5 Risk 48
3.5.5.1 Conclusion 49
3.5.6 Beliefs 50
3.5.6.1 Conclusion 50
3.5.7 Control 50
3.5.7.1 Conclusion 51
3.6 TRUST AND ITS RELATED CONCEPTS 51
3.7 TRUST AND COMPLEX SYSTEMS 52
3.8 HYPOTHESES 53
3.8.1 Introduction 53
3.8.2 Familiarity 53
3.8.3 Goals 54
3.9 CONCLUSION 56

4. MEASUREMENTS AND SUBJECT OF RESEARCH 58

4.1 INTRODUCTION 58
4.2 THE MEASUREMENTS 58
4.2.1 Introduction 58
4.2.2 Measure 1: Solution-acceptance speed (TRREACT) 59
4.2.3 Measure 2: measures of behaviour from logfile (TRBEHAV) 60
4.2.3.1 Introduction 60
4.2.3.2 Measure 2a: number of studied solutions 60
4.2.3.3 Measure 2b: changes of solutions 60
4.2.3.4 Measure 2c: qualitative information 61
4.2.3.5 Conclusion 61
4.2.4 Measure 3: measures of behaviour from the analysis of protocols (TRCATEGO) 61
4.2.4.1 Introduction 62
4.2.4.2 The observation measures 62
4.2.4.3 Conclusion 64
4.2.5 Measure 4: The questionnaire (TRQUEST) 64
4.2.5.1 Trust 64
4.2.5.2 Conclusion 65
4.2.6 Other measurements 65
4.3 WHAT DO THE FOUR MEASURES MEASURE? 66
4.4 SUBJECT OF RESEARCH 67
4.4.1 Expectations 68
4.5 TYPE OF RESEARCH 68
4.6 CASE STUDIES 69
4.7 CONCLUSION 70

5. TEST I (ZKR) 72

5.1 METHOD OF TEST I 72
5.1.1 Introduction to Test I 72
5.1.2 The stimulus material of Test I (the ZKR program) 72
5.1.2.1 Background of ZKR 72
5.1.2.2 Theoretical background to staff planning or shift scheduling 73
5.1.2.3 The ZKR 1.2 program 74
5.1.2.3.1 Administrating 75
5.1.2.3.2 Counting 76
5.1.2.3.3 Harmonising 77
5.1.2.3.4 Valuing 78
5.1.2.3.5 Adapting 78

5.1.3 The user assignments 79
5.1.4 Location of the tests 81
5.1.5 Subjects of Test I 81
5.1.6 Design and procedure of Test I 82
5.1.7 Measurements of Test I 82
5.2 RESULTS 82
5.2.1 Reliability test of the questionnaires 82
5.2.1.1 Conclusion 83
5.2.2 Normality of the four trust scales 83
5.2.2.1 Conclusion 84
5.2.3 Correlation between group and the variables: age, gender, computer expertise, computer anxiety and computer attitudes 84
5.2.4 Correlation between group membership and scales of trust 85
5.2.4.1 Introduction 85
5.2.4.2 Age 86
5.2.4.3 Gender 86
5.2.4.4 Computer expertise 87
5.2.4.5 Computer anxiety 87
5.2.4.6 Computer attitudes 88
5.2.4.7 Conclusion 88
5.2.5 Difference between technical and non-technical 88
5.2.5.1 Introduction 89
5.2.5.2 Differences in trust between the two groups (technical and non-technical) 89
5.2.6 Difference in types of trust 89
5.2.6.1 Introduction 89
5.2.6.2 Trust I 90
5.2.6.3 Trust II and III 90
5.2.6.4 Conclusion 91
5.2.7 Development of trust 91
5.2.7.1 Introduction 91
5.2.7.2 Influencing trust negatively 92
5.2.7.3 Influencing trust positively 94
5.2.7.4 Trust over a period of time 96
5.2.8 Difference in tasks and measure of trust 98
5.2.8.1 Introduction 98
5.3 CONCLUSION 100

6. TEST II (CASH) 103

6.1 INTRODUCTION 103
6.2 METHOD OF TEST II 103
6.2.1 Introduction to Test II 103
6.2.2 The stimulus material of Test I (the CASH program) 103
6.2.2.1 Background to CASH 104
6.2.2.2 Theoretical background to CASH 104
6.2.2.2.1 The patient's route 104
6.2.2.2.2 Fitting 105
6.2.2.2.3 Theoretical background to hearing aid selection 105
6.2.2.3 The CASH program 106
6.2.2.3.1 The knowledge-based system part 106
6.2.2.3.2 The user interface 106
6.2.2.3.2.1 Making an inventory 106
6.2.2.3.2.2 Selection and appreciation 109
6.2.2.4 Conclusion 110
6.2.3 Assignments for the subjects 110
6.2.4 Subjects in Test II 110
6.2.5 Locations of Test II 110
6.2.6 Design and procedure of Test II 110
6.2.7 Measurements of Test II 111
6.3 RESULTS 111
6.3.1 Reliability test of the questionnaires 111
6.3.1.1 Conclusion 111
6.3.2 Normality of the scales 112

6.3.2.1 Conclusion 112
6.3.3 Correlation between group membership and the variables: age, gender, computer expertise, computer anxiety and computer attitudes 112
6.3.4 Correlation between group membership and the scales of trust (TRBEHAV, TRCATEGO, TRREACT and TRQUEST) 113
6.3.4.1 Introduction 113
6.3.4.2 Age 114
6.3.4.3 Gender 114
6.3.4.4 Computer expertise 115
6.3.4.5 Computer anxiety 115
6.3.4.6 Computer Attitudes 116
6.3.4.7 Conclusion 116
6.3.5 Difference between the commercial and the non-commercial group 116
6.3.5.1 Introduction 116
6.3.5.2 Differences in trust between the two groups (commercial and non-commercial) 117
6.3.6 Difference in types of trust 117
6.3.6.1 Introduction 117
6.3.6.2 Trust I 117
6.3.6.3 Trust II and III 118
6.3.6.4 Conclusion 118
6.3.7 Development of trust 118
6.3.7.1 Introduction 118
6.3.7.2 Influencing trust negatively 118
6.3.7.3 Influencing trust positively 120
6.3.7.4 Trust over the course of time 121
6.3.8 Difference in tasks and measure of trust 122
6.4 CONCLUSION 123

7. CONCLUSION 125

7.1 INTRODUCTION 125
7.2 GENERALITY OF THE RESULTS 126
7.2.1 Traditional systems versus Knowledge-based system 126
7.2.2 Traditional system development versus Knowledge-based system development 127
7.2.3 Conclusion 128
7.3 APPLICABILITY OF THE RESULTS 128
7.4 BACKGROUND VERSUS GOAL 129
7.5 THE FUTURE OFFICE 129
7.6 ORGANISATIONAL IMPLICATIONS 130
7.7 FUTURE RESEARCH 130

REFERENCES 133

SAMENVATTING 139

1. Introduction

1.1 Introduction

At this time, research on the interaction between humans and the computer enjoys broad scientific and social interest. One of the reasons for this increase in research on the interaction between humans and computers (human-computer interaction) can be ascribed, among other things, to the explosive growth of the use of computers during the last 20 years. In the recent past, computers used to be extremely expensive and were used only by experts; at this moment computers are rather widespread, affordable tools that are used by almost everyone to a certain extent.

Besides the decrease in the price of computers, the capacity of the computer has increased enormously.Now, a simple home computer is more powerful than the former mainframe computers. This has two important consequences for the use of computers. First, the diversity and the number of users has grown and, secondly, the possibilities of developing software have been extended. As a result, practically all means of interaction between the user and the computer can be supported. Every user is able to obtain a reasonably good interface for the programs. However, it is not the case that the interfaces are always attuned to the requirements of the users. In other words, not all programs are as user-oriented as they should or could be (Eberts, 1994; Johnson, 1992; Shneiderman, 1992; Booth, 1989).

It can be said that research on the human-computer interaction is research on people, computers, and the way in which people and computers influence each other. The basic assumption of almost all the human-computer interaction is that the technique is adapted, in one way or another, to the user's abilities and inabilities. Thus, when the user-orientation of computer applications is considered, it should be known what this user-orientation of computer programs actually refers to. Not every one agrees on this, and for this concept different alternative definitions have been formulated, such as: *usability, user acceptance and user satisfaction.*

1.1.1 Usability

One of the most elaborated concepts is *usability*. At this moment, it is possible to have a program tested for usability, using certain criteria. The theoretical concept of usability is applied in practice (usability testing). Booth (1989: pp.109-112) states that the following usability criteria are necessary in usability testing: Usefulness, effectiveness, learnability and attitude. Booth describes these criteria as follows:

1. A useful system is one that helps people to achieve their goals.
2. A system must be effective in that a certain proportion of targeted users must be able to use the system in a number of environments, within a certain time and without too many errors.
3. A system must be learnable, in that users must be able to learn the system after a certain amount of training. Furthermore, users who do not use the system frequently must be able to re-learn the system within a certain time.
4. A system must provoke user attitude ratings, where a certain percentage expresses a positive view of the system.

It will be clear that particularly the aspect of the users' attitude in relation to computers and computer programs is an aspect that is not completely clear. What these attitudes exactly are, is in fact, not well described anywhere, and nowhere has it been unambiguously determined. In addition to the lack of a clear description of the concept of attitude, it is also a concept that is difficult to investigate. In general, the attitudes of people towards certain topics cannot be measured directly and can only be measured by identifying indicators for attitudes. These indicators can be measured and thus give an impression of people's attitudes.

The user-orientation of a program has a direct relation to the users' attitudes. It can be stated that the user-orientation experienced by a user depends, to a large extent, on certain attitudes of this user towards computers. Conversely, it can also be stated that the users' attitudes are also influenced by the method of interaction between a user and a system.

1.1.2 Attitudes

An example can clarify the influence of attitudes. Imagine you are working with a computer program and you have made a number of changes in a database; suddenly, because something turns up, you give the command to abort the program.

Subsequently, the following announcement appears on the screen:

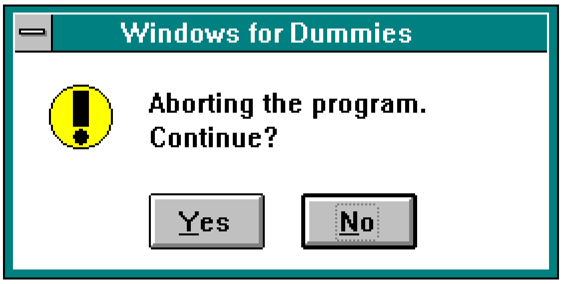

Figure 1-1: Ambiguous message

What is meant by this cryptic message[1]? Does it mean that the process of aborting the program will be continued, or will the program continue? Now you suddenly remember that you have not yet saved your data and you would like to do so before aborting the program, because the risk of losing hours of work is extremely worrying. 'Continue' could mean that you will proceed with the program, which means that you still can save the data. 'Continue' could also mean that the process of aborting will be continued, and you do not know whether you will be asked about saving the modified data. Do not forget that you may possibly have been working for several hours without saving the data. Nevertheless, you have to do something, because otherwise the program will not be aborted.

When you are used to working with the program and you have the idea that the program is user-oriented enough not to execute unexpected negative actions, then you will just make a choice without thinking too much about the consequences. However,

[1] I did come across the Dutch version of this message in a program which, of course is not called 'Windows for Dummies'.

if you do not trust the program and you have the idea that the program might start doing unpredictable things of which you might be the victim, you will often think twice, or you will try to obtain information about what this message exactly means. Within Windows you often have the possibility to ask for information (Help) concerning the various message boxes, so that you can obtain the necessary data. Furthermore, you could phone a help-desk (if there is one) or perhaps turn to another user. Whatever you do, it remains annoying for you to have to worry yourself with this problem. Whatever the outcome, your trust in the program will have decreased, and correspondingly, your appreciation will also diminish. In other words, this loss of trust will result in your attitude towards the program being influenced negatively. This one message, thus, has direct negative consequences for the users' attitudes to the whole program.

Of course such a message is ambiguous and should never occur in a program. However, there are more considerations here. Why does one user choose an option easily while another takes longer to decide on this? Why would that user trust that things will turn out well while another user does everything in her/his power to find out what is the matter and, consequently, how he/she should act? Why does one user think the same type of message in one program is more confusing than in another?
Knowing what will happen after a button has been selected solves the problem which, in principle, arises only the first time something like this happens, or when a program is not used very often. A user's reaction the first time such a message is shown depends on how the user experiences the program. This experience of the program's behaviour is the basis of the attitude a user develops regarding the program. When, working with the program, the user constantly obtains the results he/she wants and expects, then such program 'behaviour' will support the user's feeling that the program does what he/she expects of it. A user of such a program will then quickly assume that the program displaying the previously discussed message will do what the user instantly expects. The user has the feeling that the program works 'naturally' and reliably. When a user runs a program in which the actions are irreversible and in which the results are not what the user wants or expects, then the user will think longer and more deeply about his/her actions, in which process the ambiguity will be more striking, thus making it more difficult to decide. The user experiences the feeling of becoming indecisive.

This indecisiveness regarding actions that are to be undertaken can be generated in many ways. Imagine you are working with a large statistical package for the first time. You have to start executing a certain statistical test on the data. At that moment, you have the choice out of several possibilities. Which of these possibilities should you choose and, once a choice is made, will this option damage the data? Where can you find the option you want to have executed? The problem here is that you are not familiar with every option, and that the number of possibilities, which the package provides, is large. This complexity is extremely annoying and makes you uncertain about what to do. When you get to know a bit about the program and nothing unpleasant happens, you might feel free enough to just try to execute something. When you do not have this feeling, you will possibly try to read the manual in order to improve your knowledge about the program, or you could try to find a person who is well acquainted with the program so that he/she can tell you what to do next. In short, you will try to obtain control over a situation, which in principle is too complex.

Now, imagine you have made a choice and have also implemented this choice in the statistical program, after which the program informs you that the actions have been processed. Then the program shows you the results on the screen. Imagine that these results are quite important for a meeting that is to be held soon. In general, statistical programs are not known for their clarity of data processing. During the coming meeting, it will be important that you can provide, for instance, conclusions regarding the planning of the future policy of a company. Much depends on the results presented by you. Will you trust that the results will be well processed immediately and that the required statistical analyses you asked for have been carried out, or will you check whether or not everything is correct? In other words: do you rely on the program to do what you want it to do, and that it will do just what you expect it to do? Trusting that the program will work correctly and execute everything just as you expect can spare you a lot of time and discomfort because you can assume that everything has been implemented correctly - you will not have to check everything again.

1.1.3 Knowledge-based systems

In the previous section, the central issue covered was whether or not the program does what the user expects it to do. This question arises especially when dealing with knowledge-based systems. Knowledge-based systems are used in rather unstructured problem situations, those in which the solution is not obvious. A knowledge-based system has been developed for solving problems which, in the past, were solved by people. The task of a knowledge-based system is to generate a solution to a difficult problem, on the basis of input and/or data from a database. This solution to the problem is new to the user. The solution is not simply derived from the data that have been entered. Therefore it can be said that a user generally gets to see new data that have been generated by the knowledge-based system, in the form of the solution to a certain problem. In almost all cases, the user who consults a somewhat complicated system gets a solution that the user could not have easily predicted on the basis of the data entered. This solution is new and, to a large extent, unpredictable because of the complexity of the problem, the large amount of input data or the complex rules of the system. Because the user does not exactly know what happens in the system, he/she does not know whether or not the solution is correct. The complexity also means that the user cannot quickly check whether the solution has been generated in the right way and thus is qualitatively good enough. Therefore, the user has to trust that the solution that has been generated is an adequate one.

When, for instance, the user works with a knowledge-based system that, for example, schedules staff, the system must have at its disposal a large amount of facts about the availability of staff, the functions of the staff, the labour percentage, etc. Subsequently, the system should consider each of these facts, in conjunction with the rules and the inference-mechanism, in order to create a schedule. It will be clear that most users have no idea of all the rules that are used in the system, nor do they have any idea of which facts should be taken into account, and in which way, when deciding on a schedule. The generated solution is new to the user and appears not to be a direct conclusion from the simple basic data. The user expects that the system will provide a good schedule, that the rules will be correct, and that the data, which are entered by the user, will be taken into account correctly when making a decision. These are three expectations which are closely connected, because correct rules which correctly take the facts into account should result in correct schedules.

Why then are knowledge-based systems developed when it is difficult for the user to comprehend what a knowledge-based system does? The creation of a knowledge-based system is a way of reducing the complexity of problem solving. Suppose someone has to execute a task that is complex. A method of simplifying the task is to develop a knowledge-based system that supports generating a solution to this task. A result of this will be that at least a part of the complexity of the task will remain hidden to the user, because certain parts of the task will be executed by the knowledge-based system. In this way, a person who interacts with a complex system will assign a part of the task to the system. Thus, the complexity of generating a solution will be transformed into the smaller complexity of working with a complex system. By making a complex problem manageable by means of a less complex application, it becomes possible to relieve the human cognitive system. The result is that the user can solve problems with a larger complexity, as a result of the interaction with the knowledge-based system. In other words: knowledge-based systems give a user the possibility to solve problems that, without the system, might have been too complex. The reduction in complexity of problems leads to the fact that more complex problems can be processed.

Imagine that someone uses a scheduling system (see Jorna et al., 1996; Mietus, 1994) for the generation of schedules. This means that the human cognitive system is relieved of a very complex task and, one step further, it also means that the scheduler can occupy him/herself with certain aspects of making schedules that he/she thought too complex prior to the introduction of the scheduling system. The result of applying the scheduling system is that the scheduler can deal with a larger complexity than before.

1.1.4 Feedback

The example of the statistical package did not appear out of the blue. Some packages have complete spreadsheet functionalities, an extraordinary large arsenal of statistical procedures, very many graphical output possibilities and many ways of presenting results. The overwhelming number of possibilities often astonishes people who, for instance, are well informed about the background of statistics, but are working with this type of package for the first time or do nor work with it frequently. It appears that the tools with which we work nowadays are becoming more extensive and complex and that the decisions generated by the system are becoming less transparent. Many knowledge-based systems that were presented to experts elicited questions that were directed towards what happened in the system, such as: “Does it now know all data which have been entered?” And “How does it select a hearing aid?” These questions are a result of the fact that feedback from knowledge-based systems is not complete and never can be, because, for a person, it is not helpful when, for instance, only zeros and ones occupy the screen. Therefore, the user cannot literally look into the system, so that the only possibility that remains to receive information is by interaction with the system on a higher level of information feedback. The information is obtained by feedback from the system via, for instance, the way the program screen is built up, or via error messages or explanations about a generated result. Some knowledge-based systems give only very rudimentary explanations about the reasoning processes of the knowledge-based system, and generally the explanation is inadequate or impossible to understand; occasionally the information is not available at all. Mostly the feedback is at quite a high level of abstraction. Feedback could be given, for instance, about the quality of a solution in the shape of a thermometer that on a certain scale indicates the quality of the solution.

Dzida (1987) points out that the feedback from the tools with which one was working used to be much more direct. The feedback from a hammer, for instance, was very direct and at a relatively low level of abstraction. Currently, solving a problem with the aid of software is a rather complex activity. This is mainly related to the loss of the concrete visual conceptualisation of what happens in the machine. The feedback about what happens in the machine has decreased significantly and has shifted to a high level of abstraction. This results in the user having to build up a mental model of a program on the basis of what the program provides (visually, aurally, etc.).

The problem of the indirect feedback, as presented by Dzida, is very much similar to that of Human-Human Interaction. With people, there also is no direct feedback concerning which processes are taking place inside a person. The only signals someone can go on are those called 'behaviour'. In this case, the user interface consists of, for instance, the language used, the facial muscles, the movements, etc. In relation to others, people also generate assumptions about what moves a person to do certain things. Searle (1980) calls this phenomenon ‘other minds’. With this term, Searle wants to say that a person can never be certain about what happens within another person. However, from a personal frame of reference, linked to the kind of signals the other person gives, assumptions are generated about another person’s internal processes. In knowledge-based systems, this occurs on the principle of the ‘behaviour’ of the user interface, which is the basis for the mental model of what happens inside the computer and how a person has to cope with the program.

1.1.5 Magical thinking

Building a mental model is an important element of working with a program. On the basis of a mental model, a user can ‘understand’ what the possibilities of a program are. By means of these, the user can build up certain expectations about the future behaviour of a program. Constructing this type of mental model can also have problematic elements. An example of this is the so-called ‘Magical Thinking’ which Thimbleby (1990) describes. According to Thimbleby, people are not inclined to relinquish a certain mental model, but are more likely to make it more complex and/or to extend it so that certain new data are adopted into the model. This phenomenon can result in Magical Thinking. Suppose, for instance, that a command that is entered does not work, but that, at a certain moment after the user presses the keys more firmly than necessary, the command is executed. When the user makes a causal connection between typing more firmly and the execution of a command, the user will be inclined to type more firmly more often. However, typing firmly may have no relation at all to whether or not a command is executed. However the user experiences it this way because as far as he/she knows this was the only change of circumstance, and the program did work afterwards. Perhaps the user just corrected a typing error at that moment. It is possible that the user thinks that typing more firmly and lifting both feet off the ground really make the program work better. It is clear that this can go too far. The previous scene would not have taken place if the user had had a correct mental model and had known exactly how the program works and what happens internally. However, almost no one is free of this way of thinking.

I myself for instance have saved, for quite a long time, files in the computer with the ‘Save as’ option and not with the ‘Save’ option. This was because, for programming Windows Help files, the file type has to be ‘Rich Text Format’, and the word processor had written the wrong type to the disk with the ‘Save’ command, leading to much code being lost. To be certain that the right type would be written to disk, I used the ‘Save as’ function in which I explicitly specified the ‘Rich Text Format’ option. The cause of the problem later turned out to be in loading the file and not in writing to disk, while I thought that the correct form of saving only worked with the ‘Save as’ option. The problem here was that my mental model of Word did not correspond to the working of the program. In my mental model, I assumed that the default file type is applied when the ‘Save’ option is used. A correct mental model would be that Word saves a file in the same type of file format as that in which it is opened.

1.1.6 Animism

The previous description illustrates that programs and computers are increasingly ascribed qualities which are related to human qualities. Users often speak of ‘he’ or about a computer that ‘knows’. Ascribing human qualities to inanimate objects is also called ‘animism’ (Eberts, 1994). Within Human-Computer research, the metaphor in which the computer is seen as a partner is consciously made. This is called the anthropomorphic approach (e.g. Eberts, 1994). With this, an attempt is made to view the Human-Computer Interaction (HCI) and the Human-Human Interaction (HHI) as being similar. The consequences of this are that the designs of the user interface are based on the interaction patterns and characteristics that emerge when researching the HHI. According to Shneiderman (1992), although an approach like this may possibly provide much understanding of the HCI, people should be wary of animism when designing the Human-Computer dialogue. A computer will not have the flexibility humans have in, for instance, ambiguous situations. Therefore, expectations that are not correct might arise in the user, while the computer will never be able to fulfil these. Shneiderman states that, in the initial phase, it might appear to be successful to follow this animistic approach, but that, ultimately, success really comes when the designer goes beyond these fantasies and starts applying scientific analyses.

The animism with computers, in which the computer is imagined as a sort of person or living creature, has not appeared out of the blue. Computers are becoming faster and the programs are becoming more and more intelligent, therefore the emphasis is increasingly shifting from interaction between the user and the tool to the interaction between a user and a ‘collaborator’ (Booth, 1989). This changing emphasis is of importance for the perspective of this research.
Word processors and calculators become advanced tools with which multiple human tasks can be executed. Therefore, the computer is acquiring a more ‘human’ character. Knowledge-based systems, for instance, currently display this human-like behaviour. As stated previously, knowledge-based systems are systems that execute ‘human’ tasks, which results in knowledge-based systems changing the character of the interaction.

Knowledge-based systems are used for the support of a user in difficult, often badly structured problems. This process of supporting (co-operation, collaboration) means that tasks have to be shared between the user and the computer. This means that there has to be an allocation of tasks, or, as Chignell (1993) states:

> *“Co-ordinated human and machine intelligence raises problems in allocating reasoning tasks between humans and machines....”*

This aspect of the computer system is called the division of labour (Wærn, 1989). Wærn writes that the division of labour is a delicate problem on which not enough research has yet been done, and thus not enough clear solutions have yet been generated.

1.1.7 Human-Computer co-operation

When people start designing co-operating systems, they should look well at which tasks can best be executed by which actor. Here, they should look at the strengths and weaknesses of both actors. Subsequently, a decision can be made on which actor - the human or the computer - can best execute a certain task or subtask. Not only is a good division of tasks between user and computer important, the willingness of the user to assign certain tasks to the computer is also of great importance, although it could cause problems. During the inspection of a factory, for instance, it may be possible that the inspector is supported by a computer which advises on whether or not certain pumps should be switched on. Then the inspector has to consider the signs the system gives and has to act on the basis of that advice and the signs. When an inspector does not trust the data presented by the system, he/she could inspect exactly what is happening on the spot. Naturally, the whole procedure is directed towards the inspector following the data presented by the system and thus allocating the tasks in question to the system. Questions like: *“Would the user hand over certain (sub)tasks to the computer?” “Will the user be content with the abortion of (sub)tasks?” “Does the user entrust certain (sub)tasks to the computer?*” can be asked. Particularly the last question is of importance in this research.

With the examples in the previous part, the topic of indirect trust was raised. Trust is of essential importance during the interaction process. Taking the example of the ambiguous message, we have seen that it is important that, to understand a message that is inherently problematic, the user places a certain degree of trust in the correct working of the system. In the example of the statistical package, it is of importance that, when working with a complex system, the user trusts the program enough to be confident about executing the possibilities. In the case of working with a knowledge-based system, it might be of even greater importance to trust, to a certain extent, the results of the system because, without trust, a solution generated by the system will not be accepted by the user, while generating solutions to problems is the most important characteristic of a knowledge-based system.

The aim of most knowledge-based systems is to support users in solving problems. In most cases, this comes down to suggesting a possible solution to a problem on the basis of certain input (e.g. selecting hearing aids or generating a schedule). Therefore, the programs should do what the user would like them to do; in the case of knowledge-based systems, this means solving problems. An example of this type of system is the ZKR program, which is an abbreviation for ‘ZieKenhuis Rooster’ (Hospital Schedule), a nurse-scheduling program (see Chapter 5 for a more detailed discussion of ZKR). The program generates schedules on the basis of personal details (labour percentage, education, etc.) which the user enters. The schedules generated by the program will be displayed and the user can accept one of these. It will be clear that, when a schedule is selected, it is of great importance that this schedule is of good quality. Making a schedule with pen and paper sometimes costs a nurse two days' work. Accepting a solution generated by the computer can have substantial consequences for a user when the solution appears to be wrong. The schedule has to be drawn up again, or has to be adapted drastically. With the pressure of time, this is not something that induces the scheduler to rejoice. Why then does the user trust the system's solution enough to adopt this as a good solution? The first onset to an answer is given in the following sections.

1.2 Trust

In the previous description of the use of computers, the central issue was that of whether or not the use of computers could be disadvantageous to the user. In other words, does the user *trust* the computer not to undertake negative actions? Here, trust is the keyword and it is also the main subject of the research outlined in this research.

The background of this topic grew during the development of a knowledge-based system for the selection of hearing aids for people with a hearing disorder. The resulting system was called CASH (Numan, 1991), standing for Computer Aided Selection for Hearing aids. Two versions of CASH have been developed. In de following sections, the development of CASH will be used as an example to clarify why this research has been performed.

1.2.1 CASH 1

Version 1 was developed in an expert shell and was a Windows program. This expert shell had only limited possibilities for designing graphical screens, and therefore the input of an audiogram was done by entering numbers in different Windows.
In the first version of CASH, the conversation metaphor (Norman, 1986) was used. In Human-Computer Interaction, Norman recognised two types of metaphors: the conversation metaphor and the world metaphor. During the design of the interface, either of the two different metaphors can fulfil the central function. When the conversation metaphor is central, the user and computer have a conversation about an abstract world. The world is not directly present. As Mulder, Lamain and Passchier (1992) state:

“Linguistic structures refer to objects, but are not the objects themselves.”

The contact with the object is very indirect. With the world metaphor, there is some sort of virtual reality. In the world metaphor the ‘sense’ of contact is much more direct. Within the virtual world, actions are undertaken on objects and the user immediately notices what changes in the virtual world. An example of the world metaphor approach is the so-called ‘direct manipulation’ (Shneiderman, 1992). Shneiderman describes the central idea of ‘direct manipulation’ (Shneiderman, 1992, p.183) as the visibility of relevant objects and actions; quick, reversible actions; and the exchange of the complex syntax of command language in direct manipulation of the relevant objects. During the development of an interface based on the world metaphor, one starts from the knowledge the user has of the world in order to connect the user interface to the user. The user can apply the knowledge he/she has of the world to aspects of the interface. Thus, the period of training can be reduced and the user has the feeling of ‘knowing’ the interface.

After working with it for some time it appeared that those who normally help people in the fitting of a hearing aid were not pleased with the first version of CASH, nor did they think that the results were very good. These are two different elements of the interaction between the human and the computer, namely: “*What do I think of working with this program?”* and “*How well do I think the program solves problems?*” The user who was used to fill in details in a diagram was now being asked to fill in these details as numbers. The user was asked to translate the diagram into numbers that had to be filled in. The user thus only had a list of numbers and not the diagram in order to check whether or not the hearing aids that were finally selected were satisfactory. The objects (the diagrams) were no longer directly present but had to be discussed with the aid of numbers. As stated, the first version of CASH was mainly based on the conversation metaphor.

1.2.2 CASH2

Because of the discontent with the interface of the first version of CASH, which was not very user-oriented, I rewrote the whole system in Microsoft C/C++ 7.0 which has extraordinarily versatile graphic possibilities under Windows. In the development of the second version of CASH, the world metaphor was given particular attention. The use of the world metaphor in CASH version 2 rested on displaying aspects of the interface in roughly the same way as the process used to take place in the situation without the computer. First, the audiograms have to be entered, whereupon the system has to select a hearing aid. The following figure displays an input screen of an audiogram.

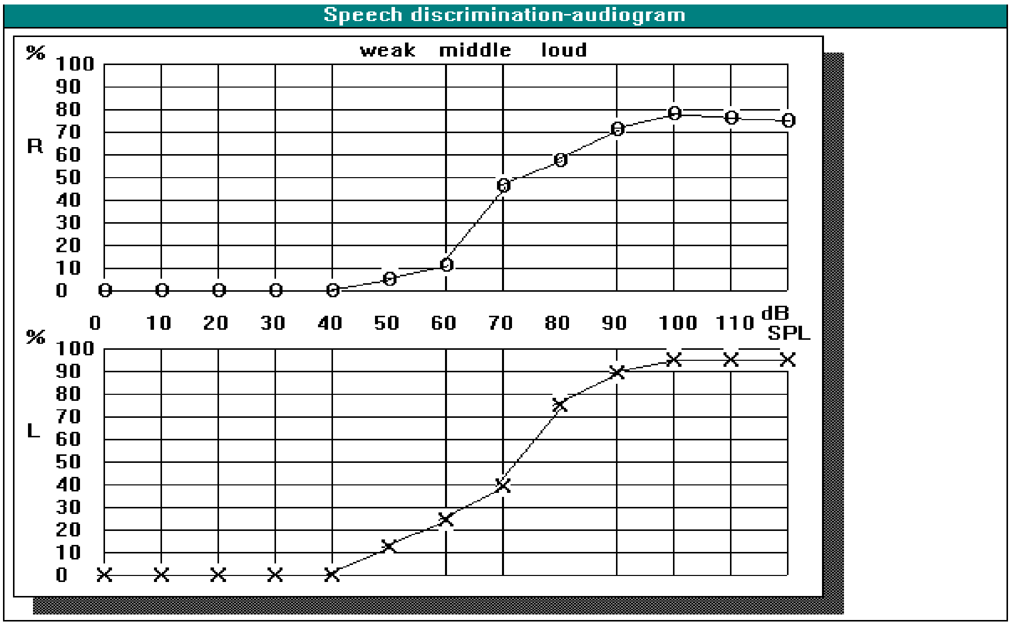

Figure 1-2: Audiogram entry window

In diagrams, which are identical to the old user-oriented diagrams on paper, a value is inserted by merely clicking the mouse button. The system uses this in order to be able to make use, in the decision process, of the value scored by the patient. Here, a clear use of the world metaphor is visible, because use is being made of old concepts that are now being used to support new task concepts. The objects in the application are a direct reflection of the objects that are used in the real world. For a more elaborate analysis, see Numan (1994).

1.2.3 CASH 1 versus CASH 2

Two significant observations were made during the introduction of the second version of CASH. First, it seemed that the system linked up well with the experts' background, and the use of version 2 was experienced much more positively than version 1. Secondly, the impression arose that not only the usage of the system was better, the experts thought the solutions provided by the system were also better. This is even more striking because, in terms of content, no changes were made in the reasoning principles of the knowledge-based system. Therefore, both the use and the generated solutions were experienced as being better than those produced by the previous version. In other words, the solutions were trustworthier. What was the cause of this increase in trust in the system? Does responding to a certain familiarity have a positive influence on trust in the program? Thus, does designing the Human-Computer dialogue influence the trust that a person has in a program? It is clear that there are factors that influence the trust a user places in a program. This means that factors that influence trust also indirectly influence the use of computers and the attitudes of users regarding software and computers.

1.2.4 Conclusion

Trust is an important construct in Human-Human Interaction. Research indicates that trust already is a major factor in Human-Human Interaction processes. In general, it can be said that trust is an essential part of the process of problem solving in groups. From the literature the following results on trust have been found:

1. Trust increases the readiness to give information (Zand, 1972; O'Reilly & Roberts, 1974, 1976; O'Reilly, 1978).
2. Trust increases the readiness to accept information from others (McGinnies & Ward, 1980; Heimovics, 1984; Early, 1986).
3. Trust has a positive influence on negotiating (Zand, 1972; Boss, 1976).
4. Trust supports participation during the process of deciding and giving advice (Hollon & Gemmil, 1977).
5. Trust supports the solving of problems in groups (Zand, 1972).
6. A positive correlation exists between trust and the motivation to bring a task to a good end (Butler & Cantrell, 1984).

As mentioned, the interaction between knowledge-based systems and users is characterised by an often far-reaching allocation of tasks. When a user consults a knowledge-based system, this user mostly wants to generate a solution to a problem (e.g. generating a schedule for staff or specifying a diagnosis). In co-operation with a knowledge-based system, the user will assign certain tasks to the knowledge-based system and will execute other tasks himself. When a knowledge-based system has been developed, the developers intend the program to be used for solving certain problems. During the development of a knowledge-based system, there is generally a difficult, quite expensive development period, and the development costs have to be repaid by the use of the knowledge-based system. If the user does not trust the knowledge-based system regarding the execution of (sub)tasks, the user will be tempted to execute the (sub)tasks him/herself. This was certainly not the intention, and the invested time and money can be seen as wasted. It is essential that people actually use the system, and fostering trust in the system can support this process. Before trust can be studied it has to be clear what trust is, how it can be measured and how it can be influenced. These topics are the basis for the research questions that will be addressed in the following section.

1.3 Research questions

In the previous section, it became clear that trust is an aspect of great importance in the research on Human-Computer Interaction, because trust plays an important role in the acceptance of the system and in the user-orientation of the applications. This means that during the development of a computer-application, one must pay attention to the trust a user has, and is able to build up, in the application to be developed. Before trust can be taken into account, it should be clear what trust exactly is. In other words: is it possible to develop a model of trust which indicates what people's trust in a computer-application actually consists of? The following issues can now be distinguished:

When trust is an important aspect in Human-Computer Interaction, what then is this trust, is it possible to formulate a model of trust, and which scale can be identified to measure trust?

We can approach these issues one by one:

1a: What is trust?
1b: What is the function of trust?
1c: Is it possible to formulate a model of trust?
1d: Which scale can be identified to measure trust?

In this introductory chapter, it has become clear that trust is a dynamic concept and that a person's trust in certain actors with which he/she interacts depends on the actors' behaviour. This means that the behaviour of an actor can support the development of a person's trust. For computer applications, this means that an application should be developed in such a way that, on the basis of a model of trust, the development of the user's trust is supported in the application. To be able to support this, it should first be clear how trust is built up. This leads to the following research question:

How is trust in a program built up and how can the dynamic aspect of trust be measured?

This research question can be divided into the following two parts:

2a: How is trust in a program built up?
2b: How can the dynamic aspects be measured?

When a model of trust is formulated and there is more clarity about the dynamic aspects of trust, it is possible to investigate whether the different parts of the model indeed also have influence on a person's trust. Important aspects of the model should be tested against reality. This provides the last definition of the problem:

3: Which aspects have influence on a user's trust in a system?

1.4 Overview

Janssens' (1982) model of the execution of research presents a cycle that is shown in the following figure.

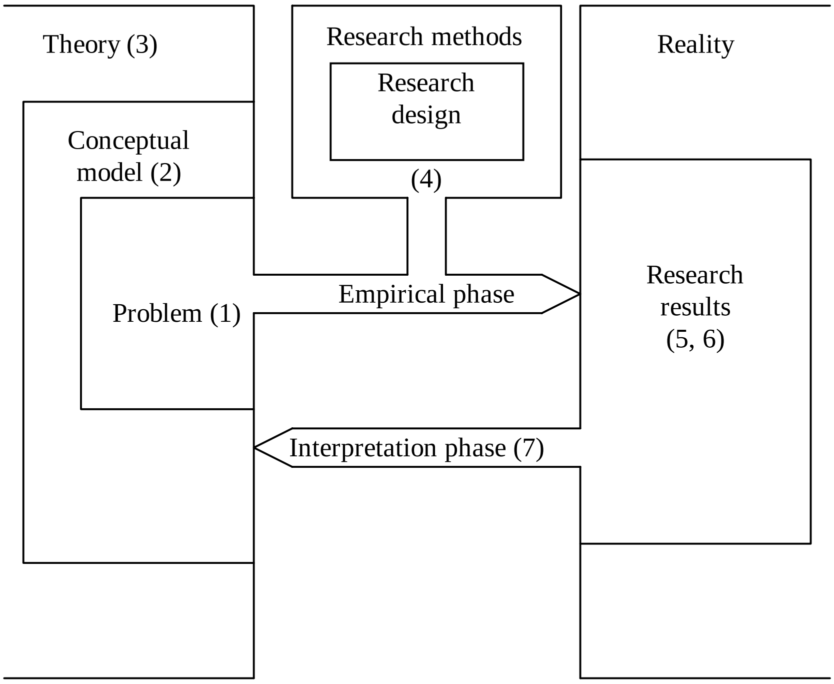

Figure 1-3: Model of the execution of research (Janssens)

The previous figure illustrates the set-up of this research. First, a problem is observed, which is then elaborated in the subsequent sections. The observation of the problem corresponds to De Groot's observation phase (see the following section). Subsequently, the theories that already exist concerning the observed problem are examined. These theories are elaborated in Chapters 2 and 3. After analysis, a conceptual model regarding the observed problem is developed and, on the basis of the conceptual model, hypotheses are formulated. The conceptual model and hypotheses are described in Chapter 3. The description of the theories and the construction of a conceptual model are, in fact, De Groot's induction phase (see following section). Then, in Chapter 4, a research design is formulated to be used for testing the hypotheses on the basis of the research methods. This phase, according to Janssens' description, corresponds to De Groot's deduction phase (see the following section). The empirical phase is discussed in the Chapters 5 and 6. This means that the results of the two empirical tests are described. This is De Groot's testing phase (see the following section). In Chapter 7, the results of the tests are interpreted and conclusions are drawn with regard to the formulated conceptual model. This last phase in Janssens' description is also the last phase in the empirical cycle (see the following section), the evaluation phase.

1.4.1 Research process

In this research, the empirical cycle (De Groot, 1963; Meerling, 1984) is used. The empirical cycle consists of five stages: Observation, Induction, Deduction, Testing and Evaluation. The cycle is represented in the following figure.

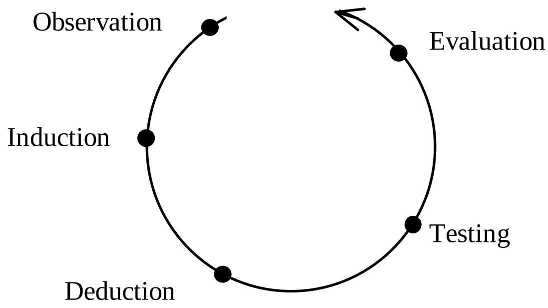

Figure 1-4: The empirical cycle (de Groot)

In these five stages, De Groot indicates how research should be correctly implemented. The five stages have the following meaning:

1. Observation: In this stage, empirical facts are gathered and the basis of the hypotheses is formed.
2. Induction: In this stage, the hypotheses are formed on the basis of the previous stage.
3. Deduction: The consequences of the hypotheses are worked out in the form of testable predictions.
4. Testing: The predictions are tested against new empirical material.
5. Evaluation: The results of the testing are related to the predictions and the hypotheses are either accepted or rejected. New theories and/or hypotheses are formulated.

The term induction is used here in the broader sense of the word, namely, going from the exceptional to the general. Therefore, in this case, deduction is mainly putting the different concepts and hypotheses into operation.

1.5 Conclusion

In this chapter, two different aspects, which play a part in computer usage, are mentioned. The first aspect mainly relates to concepts from research on Human-Computer Interaction, while the second aspect mainly refers to trust and concepts related to trust. The following two chapters are aimed at clarifying these concepts. In Chapter 2, the present state of affairs in research on Human-Computer Interaction is examined in more detail, and it is investigated made how the Human-Computer Interaction is currently being approached. In Chapter 3, the aspect of trust and its related concepts, as described in the literature, are dealt with. Furthermore, definitions are given of those concepts related to trust which are of importance in this research. A model of trust is also given in Chapter 3.

2. Overview of Human-Computer Interaction

2.1 Theoretical background to Human-Computer Interaction Research

2.1.1 Introduction

Presently, the design of the interaction component is becoming an increasingly important part in the designing process of computer programs. Users want programs that are easy to handle. Therefore, more and more research on Human-Computer Interaction is being performed, and an increasing diversity of approaches has come into being, whose aim is to design, as well as possible, the interaction component between the human and the computer. These theoretical approaches can be divided as follows:

- the task-oriented approach,
- the machine-oriented approach,
- the user-oriented approach,
- the social (context)-oriented approach.

In the development of applications, it is important to pay enough attention to the interaction component in order to be able to design user-oriented systems. The above approaches indicate different ways of supporting the design and development of the Human-Computer Interaction component. In the design and development of the Human-Computer Interaction component, it is important that each of the four approaches receives sufficient attention.

In this research, the terms ‘user interface’ and ‘Human-Computer Interaction component’ are interchangeable. Here, the term 'Human-Computer Interaction component' is mainly used, because the term ‘user interface’ is mostly only applied to the screen, although further on in this chapter it is made clear that 'user interface' is more than only the screen.

In order to support the design of the Human-Computer Interaction component, it should first be clear for what kind of task the system is being developed. Therefore, the emphasis is on the execution of tasks, which the task-oriented approach deals with. The task-oriented approach is mainly directed towards the task that has to be supported by means of a computer system. Without understanding the task, which has to be executed, it is impossible to design a system that is going to support it. When it is clear which task has to be executed, it should become clear which parts of the task have to be executed by the machine and which parts have to be executed by the user. This means that, in the design, both the machine and the user have to be examined. These approaches are respectively called ‘the machine-oriented’ approach and the ‘user-oriented’ approach.

The machine-oriented approach is mainly directed towards those aspects (possibilities and impossibilities) of the machine which are of importance when designing the Human-Computer Interaction component.

The user-oriented approach is mainly directed towards those aspects (possibilities and limitations) of the user which are of importance when designing the Human-Computer Interaction component.

At this point it is clear which task has to be supported and it is also clear which (sub)task the machine or user is going to execute. When it is known who, in principle, will execute which (sub)tasks, a system can be built. However, the problem with this is that a computer system is not used in isolation, it is used in a social context. The social (context)-oriented approach should provide understanding of how the social context is influenced by the introduction of the system and how the social context influences the development of a system.

The above approaches are all more or less taken into account in the development of the Human-Computer Interaction component. However, in most cases there is more emphasis on one of the approaches, although all are of essential importance. Each individual approach has its own strengths and qualities, while none of them can survive without the others. In the development of the Human-Computer Interaction component, it is essential that all four approaches be used.

In the following section, these approaches are examined on the basis of the ARK-G system. The ARK-G system is used for the experiment in this section to clarify the design and development process; it was not used in this research.

2.1.2 The ARK knowledge-based system

ARK-G (abbreviation for *Artsen Roostering Kindergeneeskunde-Generiek*, Generic Scheduling for Doctors in Paediatrics) is a knowledge-based system which was developed for scheduling doctors on the paediatric wards. The system was developed with the aid of two scheduling experts from the paediatrics ward. The system schedules doctors for each day, on the basis of data about the staff and the availability of doctors. The elaboration of the different approaches will be done on the basis of the development of this system.

In the project in which the knowledge-based system ARK-G was developed, special attention was paid to the aspect of separating the interface from the internal part of the knowledge-based system. A knowledge-based system (Chignell & Parsaye, 1988; Cleal & Heaton, 1988; Lucas & Gaag, 1988) mostly consists of four elements: the reasoning mechanism or inference mechanism, the knowledge-base and the database (which together form the knowledge-based system’s kernel) and the user interface (figure 2-1). The inference mechanism takes care of the reasoning of the system, such as forward and backward chaining. An important aspect of most knowledge-based systems is that the inference mechanism is separated from the knowledge-base. The knowledge-base is mostly based on ‘if...then....(else)....’ rules, that are evaluated depending on the adjustment of the inference mechanism.

The database consists of facts that are used for reasoning and for evaluating the condition part of the rules. The database can, for instance, consist of details concerning the staff, like names, addresses, etc. The user interface is the ‘look and feel’ of the program. What is regarded as the user interface in this research will be explained further on. It is important that a knowledge-based system carries out the reasoning process itself. This means that, on the basis of what is in the database and the knowledge-base, coupled to the user’s input of data, a certain reasoning or inference is followed. The result is that, to a user, a knowledge-based system obtains certain ‘human’ characteristics.

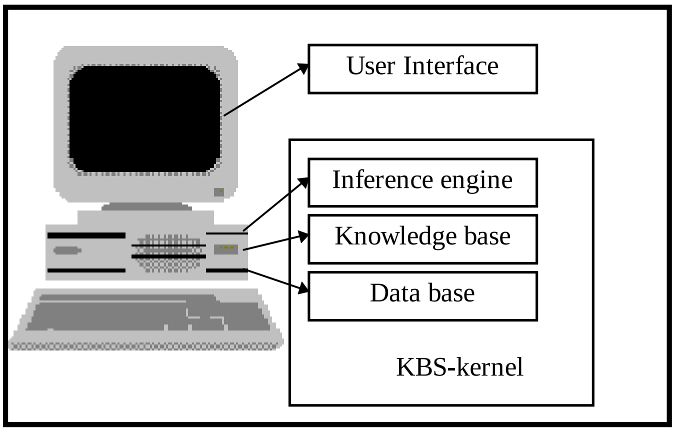

Figure 2-1: knowledge-based system

In the traditional development methods, the knowledge-based system kernel and the user interface are developed separately and are coupled to one another at the end of the course of development. Normally the knowledge-based system kernel is implemented first, and the user interface is developed afterwards. ARK-G can be divided into a part which is written in a knowledge-based system shell, and the user interface part, in which the user interface is used for knowledge acquisition (for a more elaborated analysis of this method of development, see Numan, Jorna & Arion, 1994). The separation of the interface and the knowledge-based system kernel is an aspect of the machine-oriented approach to the development of interfaces.

In order to know exactly which task has to be supported, we first looked at the problem from a task-oriented point of view.

2.1.3 The task-oriented approach

A feature of the task-oriented approach is that the emphasis is mainly on the execution of a task by a user, in which the task is the qualifying factor for the system. Research has been performed in the field of the task-oriented approach and various models have been developed. One of the first of these was the ‘Hierarchical Task Analysis’ (HTA), developed by Annett et al. in 1971. HTA divides a task into the goals, subtasks, subgoals and procedures required to achieve the goals. HTA is therefore aimed at the deconstruction of the task. A more recent method is the ‘Task Analysis for Knowledge Description’ (TAKD) by Diaper and Johnson (1989). TAKD identified the necessary knowledge of the tasks on a generic level, through which the description of the knowledge became independent of the constraints of the task analyzed. In other words, TAKD analyzed tasks in terms of actions and objects, and identified generic actions and objects of which it was thought that the level of description made them independent of the technology and the task in which they were being observed.

In the development of the Human-Computer Interaction component, task analysis is of the utmost importance because, when working with a computer, there is the matter of the execution of a task. People use computers to help them with the execution of tasks. A good task analysis can therefore support the development of the Human-Computer Interaction component, because one obtains understanding of the functions that should be supported in order to execute a task. In the ARK knowledge-based system, task analysis was mainly done during the knowledge acquisition phase. The tasks were divided into subtasks. In the development of ARK-G, it was already known which tasks and subtasks had to be supported. However, how this task had to be supported and which task the machine (computer) could have in this was not exactly known at that moment. Therefore it became important to investigate which parts could be executed by the system and which parts had to be executed by the user.

2.1.3.1 Conclusion

At this point a shortcoming of the task-oriented approach became evident. Using only the task-based approach to design a system is not enough. It is not enough to know which (sub)tasks have to be executed. It is at least just as important to know which (sub)tasks have to be executed by the computer and which have to be executed by the user. On the basis of what is known about the possibilities of a computer and the capabilities of people, we can look at which tasks are executed best by the computer and which are executed best by the user.
However, it is also important that the exercise of identifying the different (sub)tasks is carried out, with the aid of the machine- and user-oriented approaches, because this is the basis of the elaboration of the specification. Therefore, in the following section, the machine-oriented approach is investigated, and then, in the subsequent section, the user-oriented approach will be examined.

2.1.4 The machine-oriented approach

The machine-oriented approach is regarded here in two different ways. The first way in which this approach can be employed is by emphasising the description of the program, the code or the tools used in order to develop the program. Here, the emphasis lies in giving application specifications. In this, a description is given of, for instance, how the application can be built up using different existing tools, or an investigation is made into whether or not there is a valid construction of a good code for the user interface. Various books have been written about the machine- or the application-oriented approach. In his research, Versendaal (1991) goes into the possibilities of separating the user interface and the application. Here, he hardly discusses the user, and tries to make some sort of shell which a programmer can use for the development of sound interfaces. Versendaal explains this on the basis of some programmed examples. This shell could be called a User Interface Management System. A User Interface Management System (Meyers, 1988) is a system in which functions have been programmed which enable a programmer to develop interfaces for applications. Therefore, the emphasis is mainly on the program that is to be developed and the code behind it or the programs used.
The second way in which the machine-oriented approach can be employed is by emphasising the system specifications. Thimbleby (1990) takes up a kind of in-between position between the user-oriented approach and the machine-oriented approach. Thimbleby (1990) discusses the discrepancy between the abilities of a computer and the requirements of a good program, starting from a user’s abilities and limitations. Here, part of the emphasis is on the possibilities and impossibilities of the various systems.

Therefore, in the ARK-G, a functional design was made of the system, which made clear which actions the interface was to execute, among other things, and which actions had to be executed by the underlying knowledge-based system shell, based on the first knowledge acquisition round. This meant that the knowledge-based system kernel was separated from the user interface and that for both elements a clear task division and way of communication had to be designed.

2.1.4.1 Conclusion

The machine-oriented approach, thus, says nothing about how exactly the interface should look like nor exactly which tasks have to be executed by the system. A specification on the basis of the machine-oriented approach alone is not enough to design the Human-Computer Interaction component. The machine-oriented approach does not provide enough understanding of the interaction process to allow the design of a complete Human-Computer Interaction component. For the design of a Human-Computer Interaction component, it is important to know in advance what a system is and is not capable of. Unfortunately it is not enough - it is also necessary to say more about the future users.

Until now, in the example of the development of the ARK-G system, the system requirements have been worked out based on the task that has to be executed and the various software parts. A knowledge-based system should work in co-operation with a user, as was mentioned previously. This far, the aspect of the user has not been dealt with. In the following section, the user who has to work with a system is investigated in relation to designing the Human-Computer Interaction component. This is called the user-oriented approach.

2.1.5 The user-oriented approach

The users-oriented approach is a way of looking at the development of the Human-Computer Interaction component, which is based on the user with all his/her abilities and limitations, both cognitive and physical. A large amount of literature is available on the topic of the user-oriented approach. Various ways of modelling the user have been developed. Moran (1981) developed a modelling technique which is called the ‘Command Language Grammar’ (CLG). CLG is a model that describes the interaction from the input and output of the computer system. The CLG divides the interaction into different layers: the task level, the semantic level, the syntactical level and the interaction level. On the task level, the computer system is described in relation to what the system has to do, without going into the computer components or the interface in detail. On the semantic level, the computer objects and the possible manipulations/actions/operations are described. The syntactical level describes the structure of the language in which the user has to communicate with the system. The interaction level describes the system on the basis of the physical actions that are executed by both the user and the system in order to communicate.

Card, Moran and Newell (1983) developed the so-called GOMS method. GOMS is short for Goals, Operators, Methods and Selection rules. Goals are symbolic structures that define a goal. Operators are elementary perceptual, motor or cognitive conducts which are necessary to influence the mental state or the task environment. Methods are descriptions of procedures to achieve a goal. This is one of the ways of being able to save knowledge of a certain task. Selection rules have a relation with situations in which more than one method is possible in order to reach an aim. Selection rules take care of selecting the right method. Later models, such as Payne and Green’s (1986) Task Action Grammar (TAG) and Kieras and Polsons’ (1986) Cognitive Complexity Theory (CCT) are, in principle, extensions of the earlier modelling techniques.

Another way of looking at the user can be found in the ergonomic and cognitive psychological approach. Shneiderman (1992) founds his theories on cognitive and ergonomic psychology. Shneiderman (1992), Gardiner and Christie (1987), and Nickerson (1986) give a survey of techniques, methods and ideas which designers could use during the introduction of the ergonomic aspect when designing interfaces. It is important that, in this way of designing the interaction, the user is the focal point. Here, an attempt is made to design the interaction component in such a way that it is as suitable as possible for the user, based on the cognitive and ergonomic principles.

An example of the user-oriented approach to developing is the use of what the future user already knows, in other words the use of the world metaphor which was discussed in Chapter 1. With ARK-G, a part of the knowledge acquisition was directed towards analysing the scheduler's task, in which the scheduler executed the scheduling task and expressed aloud what came to his/her mind. These 'think aloud protocols' were the basis for an analysis of the development of the knowledge-based system kernel. By looking at what the schedulers do in practice, it became possible to create an interface that was closely linked to the scheduler's perception.
For ARK-G, a protocol was made first, based on the way in which the future users were accustomed to visualise the results of the scheduling process. This meant that the solutions of the system were shown to the user as a list of names for a scheduled week. The emphasis, thus, is on the representation of the results.

Because the character of scheduling was strongly directed towards running through lists of persons, it was decided to base the surveys of the schedules generated by the system on the list of persons as well. The users were shown a list of persons and could see a person's schedule by clicking the mouse on this person's name. This prototype was called ARK-X (X is for eXtra). ARK-X, thus, was a prototype completely based on a structure of lists. During the testing of ARK-X, it appeared that the users had difficulties with this structure of lists because they could not survey the schedule. The structure of the lists was workable for the representation of the generated schedules but, for a complete survey, this structure of lists was not suitable. This meant that the lists were used to communicate about a schedule. Thus, in this case, there was a design based more on the use of the communication metaphor. With this, communication was made about the schedules that were formerly formulated in a matrix by means of lists of results. The world, in this case the matrix, was not directly represented in the interface.
When it became clear that the ARK-X interface did not work well, the other world metaphor was introduced to show the representation of the schedules. The use of this metaphor resulted in the final system that was called ARK-G.
The schedulers were accustomed to making a schedule on a planning board in the shape of a matrix with colours for the sort of shift. It was now obvious to use this matrix, which the schedulers were familiar with, as the basis of the world metaphor for the user interface. Thus, based on the matrix which experts already used, a screen was developed which was closely allied to the original form of the matrix. The matrix cells used to be coloured in with coloured plastic blocks, and each colour represented another shift. The days of the week were shown horizontally and the different doctors vertically. Filling-in shifts were used to be done by choosing a block and placing this in the matrix on an intersection of a day and a person. This way of working was also chosen with the Human-Computer Interaction component. First, the user uses the

mouse to select a kind of shift in a separate window and then clicks (places) the selected shift on its place by clicking with the mouse in the matrix. It appeared that this interface linked up better with the user's perception than the interface that was based on the lists.

2.1.5.1 Conclusion

The user-oriented approach is an essential way of looking at the interaction component. Nevertheless, this approach cannot stand on its own. If one has not performed a good task analysis, and when it is also not known which (sub)tasks are best executed by the system or the user, then it is of almost no use to look towards the user of the future system. Thus, it appears that the previous approaches cannot stand on their own, but they complement each other well on essential points. In other words, each approach complements the others on crucial points and none of the approaches, taken alone, is sufficient to be able to develop the Human-Computer Interaction component well.

In the previous sections, the task, the machine and the user aspects of developing the Human-Computer Interaction component were clarified. However, of course, a system and its user are linked to their environment and particularly the user will be strongly influenced by his/her direct environment. In the following section, the aspect of the social context of the interaction with the computer system will be dealt with in more detail.

2.1.6 The (social) context-oriented approach

In the previous sections, the emphasis was mainly on the user, the task and the computer system. Of course, it is a fact that the interaction between a user and the computer system does not occur in a void; the interaction takes place in a certain (social) context. This context is of influence on the interaction between the user and the computer system. Therefore, the need arises not only to approach the Human-Computer Interaction component from the basis of an isolated interaction between a user and a computer system, but also to consider the influence of the environment in the development of the Human-Computer Interaction component, and the influence of the computer system on the context.

Thus, the social context is regarded in two different ways: the impact of a system on the social context and the impact of the social context on the development of the Human-Computer Interaction component. In the following section, both ways will be discussed.

2.1.6.1 The impact a computer system has on the social context

Shneiderman (1992) elaborates on the consequences that the introduction of a certain computer system design can possibly have on the social aspects. Here, he sees some problems (*"Ten plagues of the information age"*) which have to be solved by the designers. He also gives solutions to overcome the problems. In addition, he gives a "*Declaration of responsibility*" in which he states that every computer system designer has certain responsibilities regarding society. Shneiderman puts the emphasis on the impact that a new system has on the (social) context. According to Shneiderman, every designer of new (computer) systems has a certain responsibility regarding the future users.

Booth and Brown (1989) have a similar approach when the social context is brought up. They mainly discuss the impact a computer system can have on individuals, work groups and organisations. They deliberate on problems like: change of tasks due to the introduction of computer systems (see also Numan, Jorna & Arion, 1994), the increase of control over individuals, and group conflicts. Nowadays, we cannot do without the computer in our society, which shows that the computer's influence is extremely extensive.

During the design of the scheduling system, for example, it is important to look at the influence the system has on the social context. Installing the system without having considered the consequences is not sensible. When, for instance, the management decides to have a system designed that schedules staff, one may suspect that the thought behind this is that the management wants to exert influence on the scheduling process. It is possible that the scheduler has indicated that there were too few members of staff on a department and that, because of this, the work pressure was too high. The management now wants a scheduling system to be developed in which the management can introduce certain preconditions for generating schedules. In this way, the management directly controls part of the scheduling process. It could be the case that, indeed, scheduling can take place more efficiently, which could mean that people have to be dismissed, and that the work pressure, which perhaps was already high will become even higher. The social implications of such a scenario will be clear.

An example of the influence on the social context of a newly introduced system emerged with the development of CASH. The system was meant to support the experts and to relieve them during the selection of hearing aids for people with a hearing disorder. During the knowledge acquisition phase, someone from the management of the audiological clinic dropped a hint that the system might replace the experts. The remark was not meant to be taken seriously but was indicative of the thoughts the staff had about the system. A consequence of this was that the experts wanted to stop co-operating. A split came in the relationship between CASH'S expert in knowledge acquisition and the clinical experts. Only after much effort had been put into repairing the mutual distrust, which had arisen because of the introduction of the new system, did the experts agree to co-operate again.

In this research, the emphasis has been placed mainly on the impact of the social context on the design of the Human-Computer Interaction component. In the following section, this approach will be elaborated further.

2.1.6.2 The impact of the social context on the design of the Human-Computer Interaction component

In the previous section, the influence of a system on the (social) context was discussed. This current section deals in more detail with the influence of the context on the design of the Human-Computer Interaction component. During the development of a scheduling system, the issue arose of whether or not there should be a password on the system for the users and the management. This question did not originate from the analysis of the schedulers' task, but came up because of the social context in which the system would have to work. Had the interaction between the human and the computer been examined without looking at the social context, this part would never have come to light, while this can be of great importance to certain applications. The development of the Human-Computer Interaction component, without taking into account the influence of the social context, can be a cause of wrong designs. Taking account of the influence of the social context is necessary to be able to make use of a theoretical framework that researches the development of the Human-Computer Interaction component.

In the field concerning the impact of the social context on the development of the Human-Computer Interaction component, various trends can be distinguished. One of the theoretical views corresponds to that expressed in the Russian psychologist Vygotzky's 'Activity Theory' (see Kaptelinin, 1992; Bødker, 1989, 1991; Bannon & Bødker, 1991). Here, a more holistic way of research on HCI is propagated. Based on this 'Activity Theory', the 'Action Theory', for instance, by Frese and Brodbeck (1989), Frese and Zapf (1993), Dzida (1987) and Roe (1988) has been developed in German industrial psychology. This approach can also be seen as a more holistic approach in which the user's situation (context) is assigned a prominent place.

Another approach which receives more influence is the so-called 'Ethnographic approach' (Kuuti & Karasti, 1994; Pycock et al., 1994; Hughes, 1993). In the ethnographic approach, there is an attempt to describe the work in terms such as the 'organisation of work', 'the flow of work' and 'the distribution of work', in which the description of the tool (read computer) is aimed at clarifying the meaning these tools receive as a result of the social and organisational context. The tool is seen as part of a work environment and therefore has to be described in these terms, too.

The social-oriented approach is the most complete way of describing the Human-Computer Interaction component; however, it is also the most vague approach and does not possess enough empirical material to support the development process adequately in practice. In conjunction with the previous approaches, the social-oriented approach can be of great value for the design of the Human-Computer Interaction component of a new system. Here too it is the case that the social-oriented approach on its own is not enough to design the Human-Computer Interaction component well.

2.1.7 Conclusion

In the previous sections, a survey has been given of the various approaches that are used in the Human-Computer research. Each approach has its own focus. Different aspects of the design of the Human-Computer Interaction component are discussed in the research on the Human-Computer Interaction. The task that has to be executed is examined, the possibilities and impossibilities of both the computer and the user are investigated and, finally, the influence from and on the social environment is analyzed. Each of these approaches is of essential importance in the design of the correct Human-Computer Interaction component.

What, however, is not being discussed very well, with the exception of some specific subjects like computer anxiety, are the specific attitudes users have regarding computers and programs. More specifically, for example, the research on the trust people have in computers has not been given a place within the four approaches. As was outlined in Chapter 1, the trust people have in computers is of essential importance for working with a computer.

The approaches discussed in the previous section mainly have a relation with reasonably clear systems possessing a reasonably predictable reaction pattern. Therefore, it becomes possible to describe the interaction with the system quite comprehensively by means of the task-oriented, the user-oriented, the system-oriented and the social-oriented approaches. A problem arises when the system starts behaving unpredictably. In Chapter 1, the complex systems, and particularly the knowledge-based system, were discussed. Here, the altering character of the various systems was looked at exhaustively. In the user's interaction with these systems, the user has the feeling that the system has received a larger 'freedom of action' which formerly was only found in intelligent actors. The result, among other things, is that, for the user, the system has acquired a larger unpredictability. The four approaches discussed say little about these characteristics.

That the user ascribes a certain freedom of action to the knowledge-based system is remarkable. However, with the knowledge-based system, another aspect emerges. A knowledge-based system is mainly used to solve problems that were formerly solved only by people. Therefore, in order to work well with a knowledge-based system, the user literally has to give it a certain freedom of action by handing over certain cognitive tasks which were formerly only executed by people. This means that the user might feel a bit lost. The system takes control of certain cognitive tasks in such a way that the user has the feeling of no longer having control of these tasks.

The current Human-Computer Interaction (HCI) approach aims at, as was stated previously, a dialogue with reasonable, statistical systems that can be understood by the user up to a certain level. Because, in the perception of the user, the present knowledge-based systems have the characteristics of an actor, this approach in terms of HCI is no longer sufficient. As was outlined in the previous sections, the traditional approach to HCI research is not sufficient to perform research on the actor characteristics that the users assign to the system. An extension of the standard HCI theories has to be made in order to deal with the actor-character of the new knowledge-based systems.

To fill up the previous deficiency in the HCI research, the factors that influence trust in knowledge-based systems can be examined by various groups of users. Therefore, one of the important questions becomes:

"Which factors influence the use of different knowledge-based systems via trust?"

Based on the previous information, the perspective of this research on Human-Computer Interaction can elucidated.

2.2 Perspective on HCI

In order to clarify what is seen as the Human-Computer Interaction within this research, a quotation from Johnson (1992) will be given:

> *"The interaction between the user and computer does indeed occur through input and output devices [....], but the user interaction is influenced by many things that are often hidden from the user's view. Consequently, the user interface is more than what is visible or touchable at the various input and output devices that the person uses. [.....] The interface between the user and computer is unclear. There is not just a physical separation of person and machine."*

Johnson (1992) states that the user interface is not only a physical separation of person and machine. However, what then is the user interface in which the interaction takes place? An unequivocal answer to this question cannot be given. However, it can be stated that the interaction is influenced by what is generally called the *look and feel* of the user interface. These look-and-feel aspects of the user interface have not been sufficiently covered in the traditional data-processing psychological approach. Carroll (1991) states in his *Kittle House Manifesto* that the application of methods of research in Human-Computer Interaction has to go beyond the data-processing psychology and maybe further than psychology in the shape of social and behaviouristic approaches. Carroll says the following about this:

> *"HCI seeks to produce user interfaces that facilitate and enrich human motivation, action and experience, but to do so deliberately it must also incorporate means of understanding user interfaces in terms of human motivation, and experience."*

Here, it is clear that aspects such as motivation, action and experience are aspects that have to be explicitly emphasised for a good understanding of the Human-Computer Interaction component. Booth (1989) has a similar approach on Human-Computer Interaction, and states that the social context is also an important aspect in designing and understanding the Human-Computer Interaction component. He states on page 94:

> *"In short, cognition does not operate in isolation and, if we are to understand the problems of HCI, it cannot be fruitfully considered outside its everyday context."*

It is clear that Human-Computer research demands a broader approach than that based only on data-processing psychology. In this research, an impulse is given towards meeting this broader approach. The subject of trust is coupled with Human-Computer Interaction. In the following figure (2-2), a schematic reproduction of the basic aspects of this research is shown. The horizontal arrows stand for Human-Computer Interaction (HCI). HCI is more than only what the user can see of a system. The top horizontal arrow represents what the user sees of the interface and the bottom arrow represents the user's expectations of the system. The top horizontal arrow is mainly aimed at the user's data-processing of the feedback from the system, while the bottom horizontal arrow stands for the expectations, motivations and ideas about the computer, which are based on the feedback from the computer.

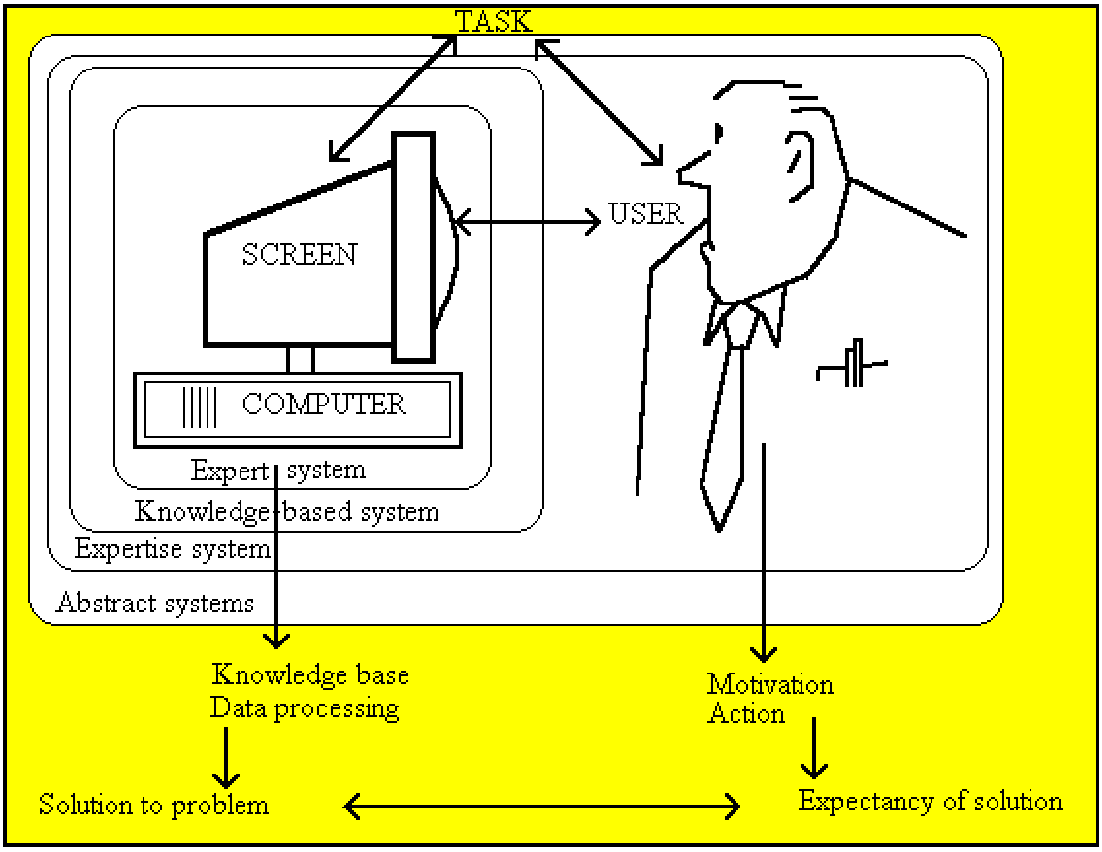

Figure 2-2: schematic reproduction of the basic aspects of this research

At the top of figure 2-2 there is the triangle: Task, Screen/Computer and User. The Task, User and Screen/Computer (system) triangle is the basis of the Human-Computer Interaction research. Within a working context, the computer (system) is used by a person (user) who executes an assignment (task). For ARK, thus, the user is the scheduler, the main task is generating a schedule and this is done with the aid of a computer system. Frese and Brodbeck (1989) distinguish a similar tripartite model, which is displayed in figure 2-3.

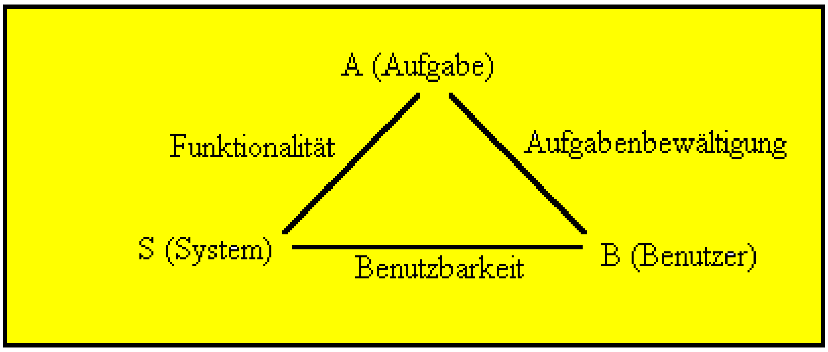

Figure 2-3: tripartite model of HCI (Frese and Brodbeck)

In the figure there are again three basic concepts: Aufgabe (Task), Benutzer (User) and System (System). In Frese and Brodbeck's (1989) figure, the three sides of the triangle are called Benutzbarkeit (usability), Funktionalität (functionality) and Aufgabebewältigung (task control). They define usability as follows: when a system is well manageable, it can be stated that there is a good usability. Functionality is defined as follows: functionality is related to whether the task which has to be executed is supported or not. The task control can be (sub)divided into primary and secondary task control. The primary task control is related to the actions, which are directly connected to the task, while the secondary task control is related to the technical aids by means of which the task is executed. An example can clarify some of this. The following figure shows the screen by means of which the starting date of a schedule is entered in the ARK schedule system.

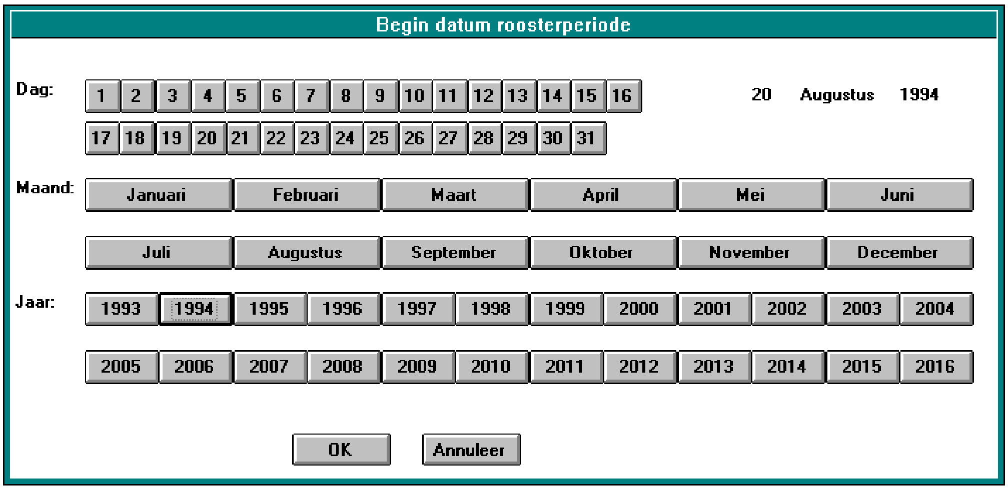

Figure 2-4: date entry window

Here the secondary task is clicking the mouse on the correct buttons in order to specify the date. This means that the secondary task elements no longer have to be executed and that these elements will be limited to a minimum. There are no extra requirements, such as typing the date in a certain way and the selection of the correct files, and consequently the user can direct his/her attention towards the important primary task.

'Task' is an important area within HCI research because, when working with a computer, as with any tool, there is the matter of the completion of the task. The system has to be adjusted in order to be able to execute a certain task, like any tool. In addition, the system has to be adjusted to the user so that he/she can use it to the optimum degree possible. Both aspects, the task completion and the adjustment to the user, are important. A problem with the development of software is that, after the introduction of a new system, the execution of tasks changes (Numan, Jorna & Arion, 1994). Figure 2-2 shows how knowledge-based systems can be placed in Giddens' (1990) theory, in which Giddens identifies expert, knowledge-based, expertise and abstract systems. See also section 3.2.1 in which Giddens' theory in relation to trust is dealt with in more detail. The knowledge-based system is an expertise system, which in its turn is an abstract system. At the bottom of Figure 2-2, the internal factors, which influence one another, are shown. The feedback is handled by the screen, but influences the user's expectations in relation to the solution of a problem, which again depends on the knowledge-base and the data-processing.

It can be stated that the user often experiences a computer as more than just a machine which does what the user wants, and that the interaction component between the user and the computer system is more than only that which is visible. We are inclined to personify the computer and, unconsciously, we assign it emotions and motives which it does not have (Howard, 1994). As was stated in Chapter 1, a computer has become a collaborative machine providing feedback at a high level of abstraction. As a result of the reduced and more abstract feedback, the user has to rely more and more on a representation of what actually happens inside a computer. The direct feedback of what happens in a system is no longer present and the user has to build up ‘trust’ on the basis of the feedback given by the program, via the ‘User Interface’. This is also the point where the trust is built up and broken down.

2.3 Conclusion

As a conclusion it can be stated that the interaction component between the Human and the Computer is more than only an interface. The interface is indeed the so-called access point but not the complete story. In the Human-Computer Interaction, there is more than merely the cognitive processing mechanisms. Emotion, motivation and the user’s expectations play an important part, in conjunction with the aspects of the task and the (social) context in which the interaction takes place. This means that more research should be done on the user’s emotional and motivational aspects. The previous section indicates that none of the approaches on its own is comprehensive enough to explain and predict the totality of aspects which play a part in the Human-Computer Interaction. Therefore, it is important to extend the cognitive psychological approach. In this research, adding the psychological aspect of ‘trust’ and the concepts related to that extends the cognitive psychological approach. It is not the intention to completely renounce the cognitive psychological approach, but rather to complement its shortcomings. In the following chapter, trust, as it is elaborated in the literature, and the concepts, which are important in the field of trust, are dealt with in more detail. Furthermore a model of trust, as it is used here, is presented.

3. Trust and Related Concepts

3.1 Introduction

In the previous chapter, the aspect of the Human-Computer dialogue was dealt with in depth. In this chapter, the concepts described in Chapter 2 will be discussed. First, a survey of the research on trust will be given. The basis of this chapter mainly consists of the assumption that, in general, people want to control their environment, which means that they try mainly to minimise the negative results of certain situations and that they want to maximise the positive outcomes. This involves an organism's ability to be master of its environment (Fisher, 1985). Here, the emphasis is on the organism's abilities. Frese (1987) expressed the view that the execution of control means that a person carries out actions that aid achieving a goal. In this research, two aspects are distinguished: the execution of control and the feeling of control. The difference between the execution of control and the feeling of control is that, with the former, the person actually executes control and, with the latter, the person might not have control at all, but believes he/she has control. With the execution of control, the control is real while, with the feeling of control, the control might be imaginary without the person explicitly noticing this. In this chapter, it will become clear that trust is a mechanism by means of which an individual can obtain a feeling of control. Research on trust can be described in two different ways. Trust can be regarded as the functionality of the concept and trust can be regarded in relation to content.

3.2 Research on trust

In this section, attention is paid to research that is being done on the construct of trust. This research is divided into two parts: the functionality of trust ('why and in what way does trust work?'), and the content of trust ('what does it consist of or what is it?').

3.2.1 Functionality of trust

In his book on trust, Luhmann (1979) wrote a survey regarding the functionality of trust. According to Luhmann, the current social complexity is so large that people are forced to develop mechanisms to effectively reduce the complexity. He also states that people do indeed possess and use these mechanisms. According to Luhmann, one of the mechanisms to reduce complexity is trust. Luhmann states that, on the basis of trust, people act as if there is only a fixed number of possibilities in the future. This results in people no longer having to investigate all the possibilities a future can have, but that they can limit themselves to investigating the expectations of the future. When someone trusts something, this person transforms the external complexity into a lower internal complexity. This is done because of the expectation (without being certain) that only a fixed number of events is possible in the future. This reduction of complexity has different consequences. Trust reduces the feeling of uncertainty. By assuming that there are only a fixed number of possible events in the future, a person feels more secure with regard to his or her acting or not acting. The person has the feeling of knowing what will happen in the future, because he/she does not take all the confusing alternatives into account when anticipating the future. Luhmann states that trust is used to lower the uncertainty regarding other people's behaviour, which is experienced as an unpredictability of changes. This increase in the feeling of secureness also gives a reduction of the feeling of risk which people have in certain situations. A person can get the feeling that the risk which has to be taken can be controlled when he/she interacts with the environment.

Let's now look at the problem of the interaction with a statistical software package (see Chapter 1), where the package generates an output on the basis of data entered into the computer by the user. To be able to generate this output, the user has to make a way through the different options, such as selecting certain variables that have to be processed and selecting a certain type of statistical processing. The user cannot see whether the correct variables are being used internally, or whether the correct statistical procedures are being executed. On the basis of the execution of the package, it is possible to get a summary impression of the statistics that were used and of the variables. To be completely certain that the output is correct, the user would have to manually check all the procedures for the variables and compare them to the package's output. Only then could a user be absolutely certain that the output is correct. It will be clear that, in such cases, the use of the package has become quite superfluous. The problem here is that the user can be uncertain about the package's output because of the complexity of the package. The acceptance of the output entails a certain risk because of the large complexity. According to Luhmann (1979), trust can function as a mechanism to take away the uncertainty, lowering the feeling of complexity, and minimising the feeling of risk. What the user can do is to expect that the output will be correct; in other words, the user trusts that the output will be correct. With this, the user makes an assumption about the future (namely: the outcome will be generated correctly), therefore he/she no longer has to investigate all the complex possibilities. Furthermore, the uncertainty is also transformed into a more certain expectation, resulting in the feeling that the risk has been reduced.

Thus, trust is a way of reducing the feeling of complexity, uncertainty and risk. Luhmann (1979) says about trust that it is a continuous feedback loop with signals that indicate whether or not the trust is justified. This point of view is similar to Giddens' (1990) ideas on building and demolishing trust. Luhmann also says that there is an object representing this trust and that trust is built up and broken down by the object's feedback. Giddens states that an abstract system has an access point where trust is built up and broken down. The object of trust is the abstract system and the feedback is given to a person at the access point. To be able to relate this to knowledge-based systems, one can say that the abstract system is the knowledge-base and the reasoning mechanism, and that the access point is the user interface. This can be represented in a figure as follows:

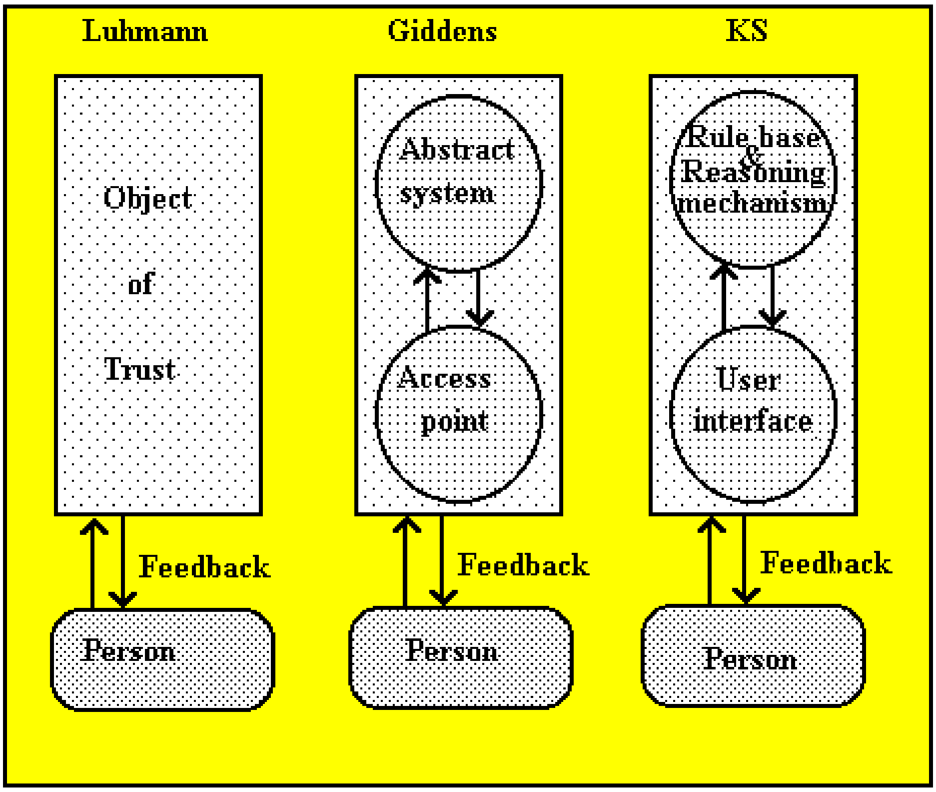

Figure 3-1: access point and user interface

Giddens (1990) states that trust is only needed when a person is not acquainted with the experts' knowledge. This means that when a person trusts something, it can be stated that there is a lack of knowledge and that there is uncertainty about something. This something can be too complex for thorough knowledge. The primary condition for requiring trust is a lack of complete information. One might say that the amount of trust is inversely proportional to the amount of knowledge. This is similar to what Muir (1987, p.532) says about trust. Therefore, when a person trusts something, he/she tries to compensate a shortage of knowledge and, at the same time, tries to reduce this complexity.

Trust is anticipating the future by assuming that the future is certain. In this way, people try to reduce the uncertainty about the future. Trust is an expectation with regard to certain aspects of an object's behaviour. This means that there has to be a certain familiarity about the object in a certain sense. Without some knowledge of the object to be trusted, no expectations can be generated regarding the future behaviour of that object. Familiarity is a precondition of trust, and familiarity and trust are linked to each other just like past and future. The past is the basis from which the future is built up. In Chapter 1, an example is given of receiving an ambiguous message. To have some trust in a correct resolution of the closing-procedure in that example, the user has to have a certain impression of the behaviour of the program in order to have an expectation of the future behaviour of the program. When the behaviour of the program is regarded as being good or positive, the user will have built up more trust in a correct resolution than when the user is hardly acquainted with the program. This history can also have fatal consequences for the use of a program. A colleague of mine, for instance, used a presentation editor under Windows and wanted to open a new document. This program can only handle one document at the time, therefore the program asked whether the current document had to be saved. While entering the name this colleague wanted to interrupt the saving process and return to the document. In general, the convention under Windows is that when the 'Cancel' button is clicked all dialogue boxes are cancelled and the whole course of action is restored to the original state. After cancelling the Save command this colleague then expected to return to the document. Unfortunately clicking the Cancel button only resulted in the Save action not being executed, while the removal of the document was executed after all. The result was the loss of a whole afternoon's work, and this colleague never used the program again. Trust needs a history of a reliable background, with which, depending on the specific history, the trust is built up and broken down. In the previous example, this colleague did not want to run the risk of the program not being reliable on other points as well, and therefore he has never used it again. Giddens (1990) states that risk and trust are intertwined, with trust normally functioning as a way of lowering the risk. When trust in a certain object is no longer present, the risk is probably experienced as large. The risk compensating working of trust has then been removed, as it was in the example just given. In this example, the feedback from the program was too small to lead to the execution of the correct action, and the trust the person had in this program, that the program would protect him from making fatal errors, was betrayed.

As stated previously, the feedback a person receives from an object he/she trusts is very important. Trust is an expectation, and when the expectation is confirmed the trust will increase, and when the expectation is not confirmed the trust will decrease. Errors that appear when working with a system could have a negative impact on the trust a person has in a system, as was shown by the example of the presentation package, while supportive feedback confirms the trust. What exactly is trust according to Giddens (1990) He defined it as follows:

> *"Trust may be defined as confidence in the reliability of a person or system, regarding a given set of outcomes or events, where that confidence expresses a faith in the probity of love of another, or in the correctness of abstract principles (technical knowledge)."*

It is at the access point, or the user interface, that the trust is built up or broken down. Therefore this is the place where expectations are influenced. Faith is some sort of blind trust, which is based on a person's ontological certainty and on a person's former experiences in new situations. Confidence is only present when a person is completely positive about certain results of certain situations. Trust, then, is only present in contingent or risky situations, when a person is aware of the possible alternatives. Trust, therefore, is a link between faith and confidence.

3.2.2 The content of trust

Muir (1987) and Lee and Moray (1992) have investigated 'trust' between the human and the machine. Two theoretical points of view, expressed in articles by Barber (1983) and Rempel, Holmes and Zanna (1985) have been taken as Muir’s basis, while Lee and Moray also involve a third point of view, that of Zuboff (1988). In the two articles (those of Muir and Lee & Moray), the authors tried to empirically determine whether it is possible to transpose theories on trust between people into theories about trust between humans and the computer.

Muir (1987) states that 'trust' has three aspects in our daily use of this concept. She states First that 'trust' is directed towards the future. There is a certain expectation of, or a trust in, someone else. Secondly there is the question of a certain referent. Someone has trust in something (a referent). In this case, a referent is a program with a user interface. Thirdly, trust is related to different aspects of a referent. These definitions are too general to operationalise, and therefore Muir (1987) and Lee and Moray (1992) took the theories discussed below as a starting point:

Barber (1983) distinguishes tree dimensions of 'trust':

1. Persistence. This means that it starts from fixed laws of nature and social laws.
2. Technical competence. This alludes to the expectation that the referent will fulfil a technically competent role. Three types are distinguished: daily routine actions, technical possibilities and expert knowledge. Muir distinguishes these three in skill-based, rule-based, and knowledge-based.
3. Fiduciary responsibility. This alludes to the expectation that a referent, at a certain moment, will allow someone else's interests to prevail above his/her own. This dimension rests less on previously shown behaviour and somewhat more on the moral obligations and intentions. When the referent's behaviour can no longer be predicted, the person falls back on this.

This point of view emphasises the static aspects of the concept 'trust'.

Rempel, Holmes and Zanna (1985) also distinguish three dimensions of 'trust':

1. Predictability. This has a relation to the consistency and desirability of previously shown behaviour. It is the basis of trust at the beginning of a relation. The more stable the referent's behaviour, the larger the trust.
2. Dependability. This has a relation to the concept of stable motives that guide the referent's behaviour. Dependability is formed by the sum of a referent's shown behaviour that indicates the extent to which the referent can be trusted in the future.
3. Faith. This is a reflection of the referent's underlying motives and intentions that form the basis of 'trust' in a relation which has already existed for a longer period of time.

These dimensions, in contrast to Barber's theory, emphasise the dynamic aspects of 'trust'.

Zuboff (1988) also distinguishes three dimensions of 'trust':

1. Leap of faith. This dimension is similar to Rempel et al.'s dimension of 'faith'.
2. Understanding. The most important aspect of understanding is anticipating future behaviour by understanding the referent's stable characteristics. This is very similar to Rempel et al.'s dimension of dependability. The dimension of understanding also involves the anticipation of future behaviour.
3. Trial-and-error experience. The basis of the trial-and-error aspect is the referent's behaviour in time. This dimension is very similar to Rempel et al.'s dimension of predictability dimension, where the basis also is the referent's behaviour in time.

Lee and Moray observe that, in these theories, the different dimensions have many similarities. In this case, 'predictability' and 'trial-and-error experience' are very similar, because they both have the referent's behaviour in time as their basis. The dimensions of 'understanding' and 'dependability' are similar, because the future behaviour is anticipated on the basis of the understanding a person has of the referent's characteristics. Rempel et al.'s dimension of 'faith' is very similar to the Zuboff's 'leap-of-faith'. The dimensions of Barber's 'fiduciary responsibility' and Rempel et al.'s 'faith' are similar because both are based on a person's expectation of the underlying motives and intentions. Rempel et al.'s dimension of 'predictability' and the dimension of 'technically competent performance' also display many likelinesses. They both rest on stable and desirable referent's behaviour.

Lee and Moray (1992) indicate the relations the three different theories have, and they state that 'trust' rests on four dimensions:

1. Foundation of trust. This dimension is exactly similar to Barber's 'persistence of natural laws' dimension.
2. Performance. This dimension is based on the expectation of consistent, stable and desirable referent's behaviour.
3. Process. This dimension rests on the concept of the referent's underlying characteristics that determine the behaviour.
4. Purpose. This dimension rests on the underlying motives or intentions. Purpose is a reflection of the designers' intentions when they designed the program.

The following figure (from Lee & Moray, 1992) presents a division of a previously mentioned dimension of 'trust'. This means that Lee and Moray have formulated a division which should completely cover the construct of 'trust', based on earlier mentioned divisions.

	Barber (1983)	Rempel, Holmes and Zanna (1985)	Zuboff (1988)
Purpose	Fiduciary responsibility	Faith	Leap of faith
Process		Dependability	Understanding
Performance	Technically competent performance	Predictability	Trial-and-error experience
Foundation	Persistence of natural laws		

Table 3-1: dimension of 'trust' (Lee and Moray)

It has become clear that trust is a construct that is difficult to investigate. Therefore, it is very tempting to split the construct. This division gives some sort of false certainty. The question is whether or not the construct of trust is still present after the division.

3.2.3 Conclusion

In the first part of this chapter, an overview of different theories about trust is presented. It appears to be difficult to give a complete definition of trust. Therefore, the construct is often divided up or is regarded at a high level of abstraction. In the first form of research (the division of the construct), it is the intention to study trust by looking at its separate parts. The problem with this is the issue of whether or not the construct of trust is still being measured. Describing the construct on a high level of abstraction gives rise to many problems about making trust operational. Therefore, it does become difficult to measure trust. In this study, trust is not split up into sub-constructs, but it is measured on the basis of a definition of trust. Furthermore, it is a serious problem that trust is linked to different constructs. In such cases, it becomes difficult to distinguish between cause and effect and their reciprocal influence. It is thus important to clarify which other aspects are important in the research on trust.

3.3 Trust

3.3.1 Introduction

Despite the previously mentioned problems, which a person can have with a complicated system, people in general are capable of interacting well with such a system. The mechanism that is used in this is that people trust certain things. If we had no trust in certain things in everyday life, we would not be able to function in current society. It has become impossible for modern humans to be completely familiar with all the elements of our society. Trust has become an essential mechanism for survival in the complex society we have built up.

Almost everyone will trust, when entering a building (assuming it is not a ruin!), that the stairs to the next floor will not collapse. This changes when we have to reach a floor via a lift and we do not trust the lift at all. This simple example indicates how disastrous a lack of trust can be in people's everyday functioning. Trust is an essential part of human cognition, reducing the complexity of all kinds of decisions. A person who simply trusts the lift has no need to consider whether he/she will use it, and therefore will not wonder about how much time climbing the stairs will cost in order to arrive at an appointment on time. As mentioned, trust leads to situations becoming less complex, and therefore the person concerned can cope with a larger amount of complicated elements. By trusting a certain actor, a person's cognitive system is relieved of those tasks which can be handed over to the actor who is to be trusted. Thus, resources of the cognitive system are freed and can be used to deal with new complex situations. Accordingly, a person's range of actions is enlarged. In other words: trust will then increase a person's freedom of action.

3.3.2 Definition of trust

On the basis of the previous outline, the following definition of trust can be given:

> "Trust is a mental action. This action is an **expectation** which person A has of an actor B - that this actor B will act positively towards the **goals** which the trusting person A has. In this, the actor B, who has to be trusted, has the freedom to harm the trusting person A. The expectation is based on *incomplete* evidence."

Two of the most important elements in this definition are the expectations and the goals that a person has. The expectations are based on the evidence that can support a certain expectation. In relation to the actor's behaviour, a person can have different expectations. The expectation that, to the person, has the best evidential value will determine the trust regarding an actor. Furthermore, of course, a person's goal is essential in trust-building. When someone expects that an actor can and will support a certain goal, it is feasible that the actor will receive the person's trust. When someone expects that an actor cannot

support this person's goals, there is no reason for this person to trust the actor. Chapter 5 describes empirical research that deals with the aspects of knowledge and evidence that determine a person's expectations. Chapter 6 provides an outline of empirical research on the goals that determine trust.
In the definition just given, the aspect of incomplete evidence is important. This is the basis on which trust rests. In the literature, two concepts directly related to trust have been found, namely faith and confidence (Hart, 1988).

3.3.3 Faith, trust and confidence

3.3.3.1 Introduction

Faith requires no evidence while, for confidence, definite evidence and/or logical operations on definite evidence are necessary. The definition that is used in this study is as follows:

> Faith is an expectation which a person A has of an actor B, that this actor B will act positively towards the goals of person A who has trust. In this, the actor B, who is to be trusted, does have the freedom to harm the trusting person A. The expectation is based on a *complete lack of evidence.*

The definition of Confidence in this study is as follows:

> Confidence is an expectation which a person A has of an actor B, that this actor B will act positively towards the goals of a person A, who has trust. In this, the actor B, who is to be trusted, does have the freedom to harm the trusting person A. The expectation is based on *definite evidence* or *logical operations on definite evidence.*

Trust lies between Faith and Confidence, and is based on partial evidence. The expectation of a future result is, in contrast to Confidence and Faith, a matter of not being absolutely certain about the future result. People generally tend to want to know about the future as much as possible. This means that people like to have a level of trust that is as high as possible, so high that it can be called Confidence.

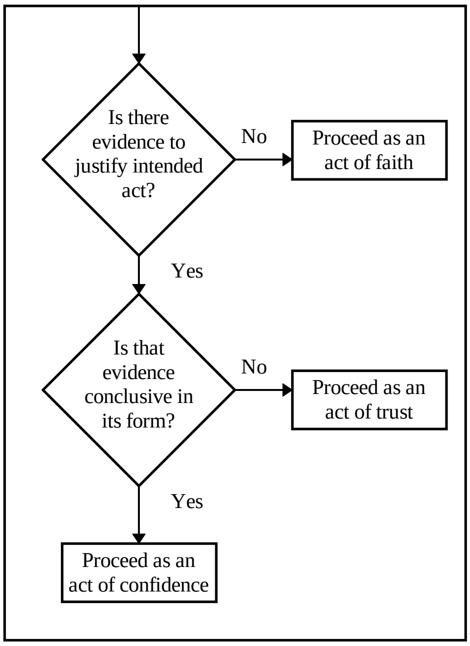

Figure 3-2: faith, trust and confidence

In the previous section, Faith, Trust and Confidence were described as elements on an evidential continuum, in which Faith and Confidence are the extremes and Trust is the link in-between. The previous figure (3-2) indicates how these three concepts are related and which criteria determine the differentiations between the three concepts.

In the above figure (3-2), the aspect of the perceived evidence is again important for a certain expectation. Whether an action is based on Faith, Trust or Confidence thus depends on how a person experiences the evidence.

3.3.3.2 Faith

When there is no evidence for the action-to-be-trusted, it is a matter of Faith. With Faith, there is no question of evidences whatsoever. Assume a person A enters a situation which he/she has never experienced before, and it appears that this person A can hand over the initiative for a certain (sub)task to an actor B with which he/she interacts. When this person A indeed leaves the initiative for a certain (sub)task to the unknown actor B, this person has no evidence whatsoever that the actor B will provide a good result after the execution of the (sub)task. At this moment we talk of Faith, and not of Trust or Confidence, towards actor B, because the action rests on a total lack of evidence.

3.3.3.3 Trust

Imagine that the person A, above, leaves a certain (sub)task to an actor B for the third time, for instance. Assume that the last two times the result was positive for person A. At this moment, person A could conclude that the two previous instances contribute to actor B's trustworthiness. Now, if the person A again leaves a (sub)task to the actor B, there is a matter of trust. It is essential that person A has the feeling that the previous results contribute to the evidence that actor B can be trusted. Here it is of importance that the evidence need not be logically correct, but that the person should have the feeling that the history says something about the future behaviour of the actor with whom he/she interacts.

3.3.3.4 Confidence

In the previous examples referring to Trust and Faith, there was mention of incomplete or even no evidence. This evidence was incomplete in terms of the person's perceptions. Now, imagine that the previous person A has the idea that actor B will do exactly he/she expects; in fact, person A also has no doubt that actor B will act according his/her expectations. At this point, thus, for person A, actor B's behaviour is completely predictable in the context that actor B will always do what is best for person A. This means that when person A has a (sub)task executed by actor B, this act rests on Confidence.

3.3.3.5 Conclusion

The above sections have shown the relation between Faith, Trust and Confidence. The limits of what can and cannot be called Trust have now become clear. In the following sections, the essence of trust will be examined.

3.3.4 The dynamics of trust

It has been stated that trust can be seen as a link between faith and confidence on the continuum of the certainty of evidence applying to expectations. When a person is in a situation of confidence, this person is in the most certain situation he/she can be. We assume that, in general, people tend to try to become as certain as possible about the future. Therefore, when someone is in a situation of trust he/she will try to reach the level of Confidence. This can be represented in figure as follows:

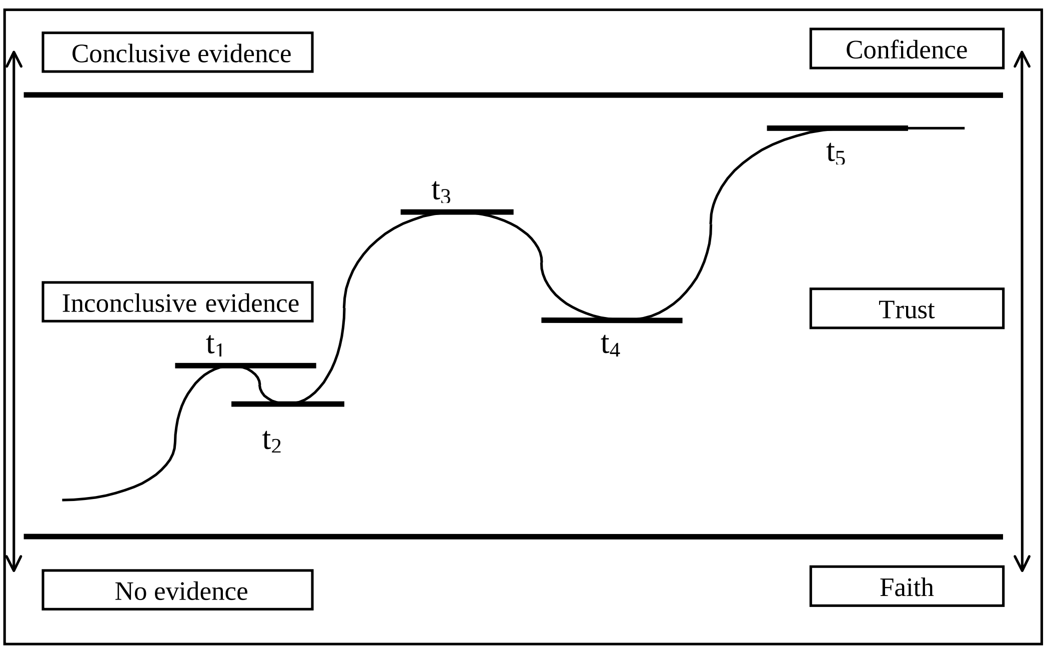

Figure 3-3: relation between faith, trust and confidence

Figure 3-3 displays how trust is strongly and less strongly influenced at different moments in time, and also shows how faith, trust and confidence are related. Time is shown on the horizontal axis, and the vertical axis presents, in principle, the amount of evidence which a person has with regard to the future positive behaviour of the object which is to be trusted. At the moments in time *t1* and *t3*, there were events that seriously damaged the trust. The trust then decreased fairly steeply. At points *t2* and *t4*, something happened which strongly supported the trust. The general tendency is that a person tries to work towards a situation of confidence. A person wants to be as certain as possible about his/her task. This will mean that, once trust is given, it is not relinquished lightly, although drastic events indeed can reduce trust.

3.4 Model of trust

3.4.1 Trust

In the following figure, a flowchart is shown in which three categories of trust have been identified:

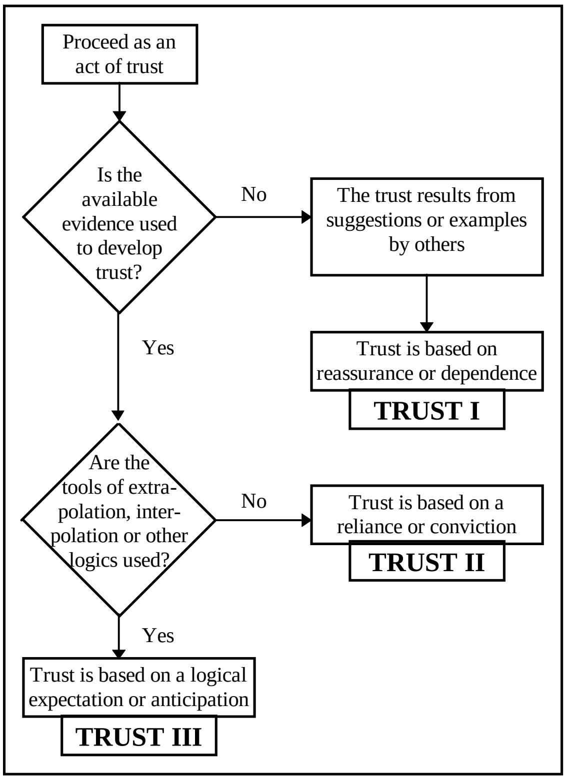

Figure 3-4: relation between Trust I, Trust II and Trust III

3.4.1.1 Trust I

The first type of trust is based on external resources. It can be based, for instance, on the fact that someone has encouraged the user to use the program. Another possibility is that a person starts to trust a program because another person is working with it and shows that he/she trusts the program. The basis of this type of trust is one or more external resources. When, for instance, someone works with a certain word processor and this person A sees someone else (B) working with another word processor, it is possible that person A might decide to trust person B's program because of the clear trust that person B radiates.

3.4.1.2 Trust II

The second type of trust is the trust which is based on empirical facts or events leading to a desired result, and with which it may be assumed that, when the conditions are the same, the result will be the same as well. Here, the conjecture is that the behaviour of the system is regarded as consistent and stable. The underlying idea is that the past and future are linked, the future being a reflection of the past. The system achieved good results in the past and therefore a person started to trust the system. This type of trust can be the result of two processes.

The first process can be described as the "trial-and-error" process that results in positive outcomes for the user. This process comes into being when, for instance, a certain goal is achieved after various attempts, and this is ascribed to the execution of certain actions with the system. In general, this process is a reliable way of working, although it can result in what is described as Magical Thinking in Chapter 1. Mostly, trial-and-error is not based on knowledge and understanding of a certain situation or system, but is based on the link of original connections between events which occur in a certain sequence. Thus, it is also possible that irrelevant actions are adopted with the idea that they are essential for a good outcome. The second process is the change of the third type of trust into the second type of trust. This will be discussed in the next section.

3.4.1.3 Trust III

The third type of trust is based on the second type of trust in combination with mechanisms that make use of extrapolation, interpolation and/or other logic. On the basis of external facts and internal mechanisms, a certain trust is built up. In the previous section, a change of the third type of trust into the second type of trust was mentioned. This change, for instance, can be the result of chunking. Someone can come to the decision to trust something on the basis of a reasoning process, after which appears that this trust was indeed justified. From now on, a chunk (meaningful unit) can be built up when the same conditions are present, and that without the reasoning process it can be assumed that the same trusting choice can be made. In this way, the third type of trust has changed into the second type.

3.4.2 Conclusion

The first two types of trust are more like automatic processes while the third type puts mechanisms into action which require more attention. In general, when people try to achieve a goal, they will try to do so in the easiest way. In this case, the easiest way mainly means that one tries to use those mechanisms that demand much attention as little as possible. In addition, the automatic mechanisms are used as much as possible. This is in accordance with people's tendency not to relinquish trust lightly once it has been invested. People tend to maintain invested trust. With the use of automatic processes, trust is only altered when a discrepancy arises between the expectations and the final result. As long as a person trusts something and the environment does not suggest that this trust is not justified, the person will continue trusting. When something happens in the environment which indicates that the trust was not justified, the mechanisms that require attention will be started up and the whole situation will have to be reconsidered. People's trust is something we can regard as being quite stable until something happens which the person had not expected, after which the whole trust has to be re-evaluated.

New relations of trust come into being when a person interacts with an actor that is too complicated, and the person has to make a choice between different options which may involve a certain risk. A person can choose between three different ways of dealing with such a problem: look for help, trial-and-error, and the use of inference mechanisms. In the case of the example of the ambiguous message in Chapter 1, a person can undertake different actions. Imagine that the person who is working with the system is afraid to lose the work he/she has just done. This person would, in order to be able to continue, go to someone whom he/she thinks has some understanding of the problem, or he/she could also try to consult the Help system. This is the aspect of asking for help (the first type of trust). Someone who can offer help can do this in two ways, by explaining the situation, and by just saying what to do. A Help system mostly gives an explanation and is aimed at generating understanding about the system. The second way of being able to solve the problem is the trial-and-error mechanism. With this, the person would simply click the 'Yes' button, after which he/she would wait and see what happens. If the person's expectations actually happen, the trust will be supported, and if something catastrophic happens, the trust will be minimised. In a third alternative, the person can start reasoning about the aspects of the program and try to extrapolate the future behaviour of the system.

3.5 Related concepts

3.5.1 Introduction

Now that the concept of trust has been examined in detail, we can take a look at the concepts that are related to trust. These concepts can be influenced by trust or trust can be influenced by these concepts. At the end of the section, the relation between the concepts and the trust will be further elaborated.

3.5.2 Complexity

The first concept that will be dealt with is the concept of complexity. Complexity has a close relation with trust. As has been stated, trust is a way of being able to cope with the complexity of everyday life. In fact, it is seen as a way of reducing the complexity of everyday life. Therefore, it is important to be able to indicate the way in which trust reduces the complexity. To this end, it is essential to first clarify what complexity actually is.

Woods (1988) identifies four dimensions of complexity of situations in the world: the measure of dynamics, the amount of interacting elements, the measure of uncertainty and the measure of risk.

The first dimension, the dynamics of the situation, is linked to aspects like pressure of time, overlapping tasks, aspects that change in time, and the measure of continuity of the task. An example of this kind of complexity is, for instance, checking an atomic plant or the production processes in a factory. The adjustment of a certain pressure in a boiler can have consequences for the pressure in another boiler that, after some time, could be too high or too low and therefore itself would require adjustment within a certain time.

The second dimension of a number of interacting elements refers to circumstances in which several/many factors are interrelated. For instance, when an error appears somewhere, this error can have more than one consequence. For example, imagine that a user of a statistical program has selected an incorrect variable for a certain statistical test. This mistake has consequences for other actions, such as the set-up of a table or a diagram. It may be possible that the data cannot be represented graphically in the incorrect variable, therefore the result of the command to present the data graphically will not be executed correctly because of the original mistake of selecting the incorrect variable.

The third dimension is the measure of uncertainty. This dimension is related to the ambiguity, the completeness or incompleteness of the information. An example of the uncertainty dimension is that of the ambiguous message in Chapter 1. The ambiguous message does not give enough information on which the user can base a subsequent decision. In other words, the user becomes uncertain about making the right decision. Furthermore, Woods (1988) indicates that the unpredictability of future situations and events has a complexity-increasing influence.

The last dimension is the dimension of risk, and is linked to the amount of risk which a certain situation or a certain sequence of actions holds for a person. The amount of risk refers to the seriousness of the consequences of a person's negative actions. It is, for example, of great importance for the person working with the statistical package in Chapter 1 that the data about the meeting are correct; therefore this user experiences the feeling of a certain amount of risk.
Dahlbom and Mathiassen (1993), however, set complexity against uncertainty. They define complexity as follows:

> *"The degree of complexity in a given situation is a measure of the amount and diversity of relevant information needed to solve the problem."*

While they define uncertainty as follows:

> *"In contrast [to the complexity], the degree of uncertainty represents the accessibility and reliability of information that is relevant in a given situation."*

What Woods sees as complexity, others see, for example, as complexity combined with uncertainty. It can be said that the opinions on complexity are not exactly similar.

In the dimensions of complexity presented by Woods, two dimensions are identified which are, in my opinion, more a result of complexity than a dimension of the construct of complexity. These dimensions are Uncertainty and Risk. Further on in this chapter, these two aspects will be discussed in more depth.

The previous Babel concerning the dimensions of complexity in this case can be ascribed to the use of the concept of complexity. Complexity is, as Frese (1987) states, something else than complicatedness. Complexity and complicatedness do overlap, but complicatedness is complexity which is difficult to control. This means that complicatedness only occurs when a situation becomes too complex to control, or when the situation is out of control for other reasons. Frese (1987) identified several aspects which make a complex problem complex, they are:
1. little functionality;
2. high loss of transparency;
3. high measure of unpredictability;
4. fewer possibilities of deciding than possible, when more decisions are needed than are mastered by one person;
5. when, both socially and technically, the complexity is not necessary.

Frese (1987) then states that complexity consists of three sets of conditions of complexity:

1. the number of goals or subgoals, plans or subplans and signals;

2. the number of relations between goals or subgoals, plans or subplans and signals;
3. the number of conditional relations between goals or subgoals, plans or subplans and signals.

These three dimensions are each divided into: time sequence, time frame and content.

Based on Woods' (1988) theory, Weir (1991) made a division of the places where complexity occurs in systems. He distinguishes three different domains of complexity:

1. complexity of the domain itself;
2. complexity of the control requirements;
3. complexity of the required user/operator interaction.

The first domain on which complexity comes to the fore is the complexity of the domain itself. This, then, is the complexity of the domain to which the complex system is connected. The complex system is meant to control a complex situation. The complexity of the domain has a direct relation to the system that has to control and maintain this domain. The amount of complexity of the domain here depends on the possible number of situations, the number of mutually connected elements and the number of possible variables.

The second domain in which the complexity of a system comes to the fore is the complexity of the control requirements. Here, the amount of complexity depends on the dynamics of the task and the control-determining factors. Here, the control-determining factors are seen as factors which also influence the control of a certain aspect of the system, such as the efficiency of a central-heating boiler which is dependent on both the water temperature and the water pressure, where the water pressure can possibly be controlled by the water temperature.

The third and last dimension (user/operator interaction) mainly rests on the appeal by the system to the user. The amount of complexity depends on the cognitive and ergonomic load and the available interaction modules. A system, which performs equally well when compared to another system on both former dimensions (complexity of the domain itself and complexity of the control requirements), can be much more stressing, for example, on cognitive and ergonomic aspects. Therefore, the system is experienced as being more complicated.

The above tripartite division is presented in the complex systems themselves. The complexity of the domain is linked to the physical processes to which the system is connected. The complexity of the control mechanism is related to the checking apparatus and the complexity of the required user/operator interaction is linked to the human-machine interface.
It is clear that complexity is something other than complicatedness. In this research, then, the complexity construct is not used; the complicatedness construct is used instead. In the following section, the aspect of complicatedness is discussed in more depth.

3.5.3 Complicatedness

Complicatedness has a large area of overlap with the previously mentioned concept of complexity. According to Frese (1987), complicatedness is complexity that is difficult to control. Frese (1987) states that a system is complicated when the system is complex and when the following conditions, linked to control, are present:

1. there is little functionality;
2. there is much non-transparency;
3. there is much unpredictability;
4. there are fewer possibilities than are necessary;
5. more decisions than a person can deal with (mental model or skill) are necessary;
6. when the complexity, both socially and technically, is inadequate.

In my view, the above six points are not all of essential importance. Point 5 is a direct result of complexity. When a person has a number of possibilities that surpass his/her cognitive capacities, he/she will think that this situation is complex.
In point six, Frese refers to a situation such as, for example, that where users used be delighted with a line editor. However, nowadays they see this as awkward and complex. I think that, here, some terms are not used correctly. First, the fact that something is given a warm welcome says nothing about the complicatedness and, secondly, when a program is regarded as being complex, this says nothing about the difference with the earlier complexity, only its relation to other programs. In my opinion, a line editor which would be used nowadays would be just as complex as a line editor that was used earlier.

Points 1 and 4 are both part of the same principle, namely, the underachievement of a system. Here, the question emerges as to whether or not a system does less than required, which can be seen as a complicating factor. Too little functionality means that a person cannot achieve a goal with the aid of a system, or that the person can only achieve a goal via an awkward roundabout route. In the first case, in my view, there is no increase of complicatedness because, when a goal is not achieved, this does not entail extra complicatedness. In the second case, the increase of complexity is the reason for the increase of complicatedness. The awkward roundabout route results in an increase in the number of goals and subgoals and thus the complexity also increases. Therefore the complicatedness is increased because this depends, among other things, on the complexity.

Thus, points 2 and 3 remain. These two aspects both rest on one and the same principle because, in both cases, the lack of knowledge causes problems. In my opinion, a lack of knowledge, together with the amount of complexity, is a primary reason for the amount of complicatedness. Due to a lack of knowledge, a user is often not able to make an adequate mental model of a system, while the amount of complexity brings about a capacity problem for the cognitive system because of the quantity of objects and relations, so an adequate mental model of the system is prevented.
When, for instance, Luhmann or Giddens talks about reduction of complexity by trust, they mean, in fact, the reduction of complicatedness.

3.5.3.1 Conclusion

A feeling of complicatedness regarding a certain situation can often generate a lack of a feeling of control. This complicatedness arises as a result of the high amount of complexity combined with a lack of knowledge about every aspect of a situation, or by the limitations of the cognitive system in being able to understand the entire complexity of a situation. Thus, the basis of experiencing a situation as being a complicated one depends on whether or not one is capable of generating an adequate mental model of a situation. The complicatedness of a situation is based on the complexity of a situation. Frese (1987) defined this as follows:

> *"A situation is complicated when there is a complex situation in which more data and decisions are required than a person (in terms of skill or mental model) can deal with. And when there is a certain degree of lack of knowledge."*

This definition means that the emphasis is on the capacities of the cognitive system. The result of this is that a person can regard a situation as being very complex, while others can regard this same situation as being relatively simple.
Another important aspect related to trust is that of uncertainty, in which trust, just as with complexity, can work as a mechanism to reduce uncertainty.

3.5.4 Uncertainty

Uncertainty arises when, for instance, a user does not know what to do, or when he/she does not know what has happened in the system. In these examples, the word 'know' is of crucial importance. In this case, the point in question is that there are problems with the knowledge with which the user reasons. For instance, a user may not know whether the knowledge that he/she has of the system is based on fact. Another possibility is that the user is offered ambiguous data, as is the case with a well-known program. This program displays a message that is ambiguous to the user, which can be compared to the example of the ambiguous message in Chapter 1. When a user wants to enter text in a screen, this input can be cancelled by means of the Cancel button. When the Cancel button is clicked, a new window appears in which the question is asked whether or not the user really wants to cancel the input. Under this question, the Cancel button is again displayed. Is this Cancel button there to affirm the command or to cancel the previous command? Another occasion for uncertainty can be, for example, when the user wants to print a file but has no idea about how this command can be given.

Uncertainty is reasoning with incomplete knowledge. This incompleteness can have three different causes, namely (Fox, 1987):

1. unreliable knowledge;
2. knowledge which is not precise;
3. incomplete knowledge.

Unreliable knowledge is knowledge in which one is not certain whether or not the data are correct. For instance, the data may be out of date, or they may have been supplied by actors of which one is not sure whether or they are working properly, such as an imprecise meter or an unreliable (re)source.
Knowledge that is not exact is knowledge that has not been supplied in the correct way. For example, taking a patient's temperature with a thermometer that measures in

intervals of 5 degrees Celsius is not precise. However under extreme circumstances, it may be possible to reason with it.
When not all the knowledge that is required for solving a certain problem is present, one can speak of incomplete knowledge. Diagnosing a patient whose historical medical data are absent is reasoning with incomplete knowledge.

3.5.4.1 Conclusion

A complicated situation can result in people becoming uncertain, because a person cannot securely estimate which actions can be undertaken. There may be a lack of knowledge or an inadequate mental model. Because of this lack of knowledge, a person does not exactly know what kind of actions can be executed, so he/she cannot be certain that eventual negative results will not turn up. Uncertainty in this research is defined as follows:

> *A feeling of uncertainty originates when a person has the feeling he/she has no adequate knowledge about the future situations in which this person might find him/herself as a result of his/her own behaviour or of that of another actor.*
> *The inadequacy of knowledge can come into being because of unreliable knowledge, knowledge that is not precise or incomplete knowledge.*

In general, people try to control situations about which a feeling of certainty arises, and they tend to prefer situations that do not bring about too much uncertainty. People try to reduce the feeling of risk and uncertainty by means of controlling the situation. In the following section, the concept of risk is dealt with in more detail.

3.5.5 Risk

A situation is experienced as being full of risk when a person expects that, in the future, he/she might eventually experience negative results that he/she cannot control, as the result of this situation.
First, it is important to know what risk actually is before it can be stated that it is an important aspect of trust. To experience a situation as being full of risk, it is necessary that certain aspects of the situation give rise to a feeling of risk. It can be said that, in this case, we are dealing with a feeling of risk, the perceived risk. The feeling of risk depends on the expected chance of a negative situation and the seriousness of the situation. A well known description of risk states:

Risk = probability of a negative situation * seriousness of the negative situation.

The emphasis is on the expectations of the person who experiences the risk.

Risk is seen as an important aspect of trust. Petermann (1992) says the following about the acquisition of trust:

> *"Wesentlich im Verständnis von Vertrauen sei: (a) der Aspect der Ungewißheit, (b) das Vorhandensein eines Risikos, (c) die mangelnde Beeinflussung des Schicksals (freiwilliger oder erzwungener Kontrollverzicht) und (d) die Zeitperspektieve (= auf die Zukunft ausgerichtet)"*

It is clear that the feeling of risk is caused by expectations regarding the future because, when negative results have already appeared, there is no longer the feeling of running a risk. Furthermore, a feeling of risk only emerges when a person has no certainty about certain aspects that may influence the outcome of a situation that was experienced as full of risk. Therefore, there are uncertain factors.

From the previous discussion, it appears that a feeling of risk comes into being when there is a possibility of negative consequences emerging from a certain situation. The feeling of risk is aimed at the future and founded on uncertainty about aspects of a situation. If all aspects of a certain situation were clear and certain, there would be no question of risk; there would be future certainty. Furthermore, risk depends on the control a person can have over a situation. When a person has total control over a situation, there is no question of a situation full of risk; this is a controlled situation. A person may depend on the goodwill of another person. In the specification by Petermann (1992), the aspect of risk is of another order than the other aspects. Risk is more likely to be a result of the other aspects taken together.

Risk, however, is an important aspect in understanding trust, as is also stated in Koller's article (1988), in which he states that risk is a determinant of trust. Koller says the following about risk:

> *"An individual perceives a situation as bearing risk if entering this situation might lead to negative consequences and if the individual is not able to control the occurrence of these consequences."*

Thus risk and trust are closely connected, and Koller even states that the amount of risk determines the required trust. Luhmann (1988) has a similar point of view and says the following:

> *[...]; trust is a solution for a specific problem of risk. [...] It [trust] presupposes a situation of risk."*

When there is an action that carries a certain amount of risk and the action is executed, then one can speak of a deed of trust. Furthermore, one can say that the larger the experienced risk, the larger the required trust in the other actor has to be, when the positive or negative outcome of this situation depends on this actor.

3.5.5.1 Conclusion

A complicated situation can create a certain feeling of risk. As outlined in this section, a person will experience a situation as bearing risk when he/she thinks it is possible, in the future, to experience negative results as a result of a complex situation over which he/she did not have control. The feeling of risk in this case is defined as follows:

> *A feeling of risk comes into being when a person has the feeling that a negative situation can lead to future situations that are negative for this person.*

Just as with uncertainty, people try, in general, to avoid situations carrying too much risk (Luhmann, 1979; Giddens, 1990). In general, some risk is stimulating, but too much risk cannot be processed well.

3.5.6 Beliefs

Until recently, the process of modelling users was mainly based on statistical models of users that were founded on information about users. Currently, it appears that these models no longer function adequately, mainly because of the appearance of 'natural language' interfaces, recognition-of-speech interfaces and co-operative systems. These systems need a more elaborate range of models in order to be usable for different users. They must have models that contain aspects like user background, beliefs, goals and plans. Ballim and Wilks (1991) add to this that the system also needs models of the user's beliefs and should anticipate these.

Ballim and Wilks (1991) state that within the Artificial Intelligence (AI), beliefs are often seen as indications of someone else's abnormalities (like: "Marc believes the earth is flat"), or remarks about things we are not certain of (like: "John thought he had placed the cup in the sink"), or aspects of faith for which no evidence has been given (like: "Bill believes in reincarnation"). Ballim and Wilks (1991), however, state that beliefs are implied by all remarks we make about the world; they define knowledge as beliefs that we think are true and for which we have enough evidence. They also say that beliefs are not reserved for individuals only but are also owned by, for instance, groups and organisations.
According to Hart (1988), beliefs are the basis of three related constructs: Faith, Trust and Confidence. The difference between these concepts is the amount of evidence that serves as the background for the beliefs. Here, Faith is not based on established beliefs but is based rather on unconditional acceptance. Trust is based on an expectation that is not unequivocally established, while Confidence is based on a strong conviction that is based on substantial evidence or logical deduction.

3.5.6.1 Conclusion

Beliefs lie at the basis of trust. Our whole trust is based on the beliefs that we have regarding the world in which we live.

3.5.7 Control

A complicated situation can give a person the feeling that he/she no longer has control over the situation. The result of a complicated situation can also be that a person experiences a lack of control.
Fisher (1985) also describes control as:

> *"[...] 'power' or 'mastery' over the environment [...]".*

Furthermore, she states that, philosophically, the concept of control is linked to the point of view that organisms try to master their environment. Fisher (1985) also says that control can be defined as the knowledge that a person can behave in a way which results in obtaining desired situations by the person. Thus, according to Fisher, the feeling of control over a situation is based on knowledge about the possible behaviour by means of which a person can influence the situation in a way that is positive for this person.

Frese (1987) gives a comparable definition of control:

> *"Experiencing control means to have an impact on the conditions and on one's activities in correspondence with some higher order goal."*

A remarkable difference can be observed between the two previous definitions. Fisher (1985) mainly aims at the possibility of control, while Frese (1987) mainly aims at the execution of control.

The previous definitions of control assume that a person exercises, or is able to exercise, direct control over the environment in order to achieve certain stated goals. This definition is no longer sufficient in the frame of the present technology and developments in society. Nowadays, people no longer have to exercise control themselves, but they can choose to have control exercised by other persons or machines such as computers. When we have a stomachache, it is possible to visit a doctor and assign our health to his or her expertise. We do not start following a course on stomachaches and how to cure these, with which we would have achieved as much control over our recovery process as possible. Instead, we hand over control to the doctor, and thus have the feeling that the doctor is exercising enough control over our recovery process. Indirectly, of course, we have control ourselves. The feeling of control over a situation has clearly increased without us exercising this control ourselves.

3.5.7.1 Conclusion

In this study, control is seen as both the potential to have control and the execution of control which can be direct or indirect. Where it is not possible to unequivocally derive whether the context requires a distinction between the execution of control and the possibility of control, this will be indicated explicitly.

Now, what are the consequences of a situation that is experienced by people as being complex? The first consequence is, as stated before, a feeling of having a lack of control over a situation that is experienced as being unpleasant. In this study, the feeling of control is defined as follows:

> *When someone has the feeling of exercising control or of being able to exercise this, he/she has the idea that he/she can minimise the negative results and can maximise the positive results of certain situations.*

Thus, the emphasis in this definition is on the feeling of control that a person has in a certain situation. This feeling of control does not have to be absolutely true, that is, have positive correlations with reality. It is possible that control cannot be exercised at all. From experiments it became clear that when people are exposed to an unpleasant situation, pressure can be dealt with better when the subjects have the feeling they can change the situation if they want to. When the subjects have the feeling they cannot control the situation, it is experienced as being very unpleasant.

3.6 Trust and its related concepts

At this point, it will become clear what is meant by trust and which related concepts are distinguished. In this section, an explanation is given of the relation between the concepts and trust itself. The following figure (3-5) represents what this mutual relation looks like.

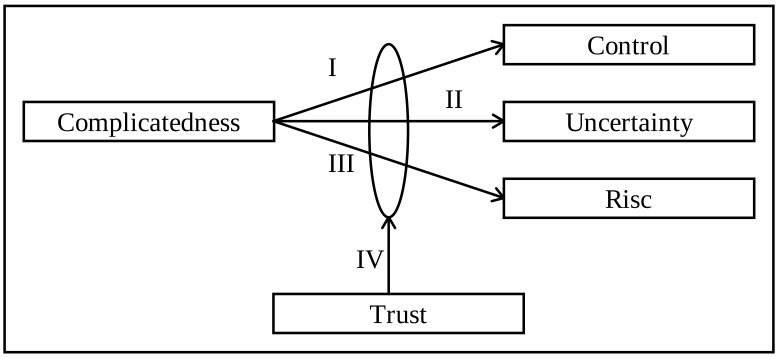

Figure 3-5: relation between trust and its related concepts

The figure indicates that complicatedness influences the feeling of control, uncertainty and risk. This implies that an increased feeling of complicatedness results in a decrease in the feeling of control (arrow I in the figure). An increased feeling of complicatedness also increases feelings of uncertainty and risk (arrows II and III in the figure). The figure also shows that trust gives an decrease in the feeling of complicatedness. This means that trust increases the feeling of control and decreases the feeling of uncertainty and risk (arrow IV in figure 3-5).

Trust is, in fact, a mechanism to compensate a lack of knowledge. The gap in our knowledge that arises because something is too complex to be completely understood is, whenever possible, filled up by the trust that we can give to another actor.

3.7 Trust and complex systems

In this section, the aspect of the interaction with complex systems will be discussed.

In this respect, a complex system can be a person or a social system or complex equipment. In this respect, a social system could be the occupational group of general practitioners or specialists. Complex equipment could be, for instance, the operating and controlling panels of a power station or a knowledge-based system. The essential feature of complex systems is that a person cannot readily fathom these. This gives the person the feeling, real or not, that the system has a certain amount of freedom to act. Only when this condition is fulfilled it can be stated that somebody is trusting the system. When a person has the feeling that he/she can fathom the system, it is not a matter of trust but of confidence, because this person is absolutely positive the system will react as the person expects.

Thus, the essence of trust, as it is used here, is the decision to hand over control over an event to a person or a system that, in this study, is called an actor. The term 'actor' expresses the idea that, to the trusting person, the system has a certain freedom of action. Therefore, in speaking of trust, it is important that there is an actor (B) that cannot be completely fathomed by the trusting person (A).

The term actor here is used in the broader sense of the word. In this respect, an actor can stand for different complex systems. An actor, for instance, can be another person, a computer or a group of persons. When we contact a specialist, in fact, we interact with a group of persons (in this case the occupational group of specialists). Here the contact plays the part of the earlier described access point. In this respect, it is important that the system, in the trusting person's view, has a certain measure of freedom of action. Thus, in this study, use is made of knowledge-based systems, which have a certain amount of complicatedness for a user, and also have a certain degree of freedom of action in the eyes of the user.

This all means that trust can be used as a concept in a broader respect than the general concept of trust between people. The big difference between trust between people and trust between a person and a non-human actor is that, in the latter case, the trust is not mutual. On the one hand, it can be said that trust becomes broader by involving non-human actors. On the other hand, this trust is less broad because trust with a non-human actor is not mutual.

At this point, all basic concepts related to trust have been defined and hypotheses can be formulated. Thus in the following sections, the hypotheses will be further elaborated.

3.8 Hypotheses

3.8.1 Introduction

The basic elements of trust are, according to the definition formulated in this chapter, expectations which a person has of another actor in relation to actions which this actor will execute in the future. A person can only have trust in an actor when a person has the goal to achieve something, and the outcome of that goal depends on the behaviour of the actor to be trusted. These expectations are always related to the goals a person has. The fact that trust has to be based on a certain degree of familiarity with, or knowledge about, an actor, which, in turn, is always based on previous experiences (knowledge) of an actor, has already been dealt with in depth. These two elements of the definition (familiarity and goals) will be dealt with a little further on. In order to be able to validate the previously formulated model of trust, manipulations by means of these elements should indicate the level of trust in a certain actor. In the previous section, it became clear that a knowledge-based system can be seen as an actor. The study has been carried out with the aid of a computer and with knowledge-based systems.

3.8.2 Familiarity

Expectations are generated on the basis of previously gained experiences that are saved in the form of knowledge and expectations. In principle, no statements can be made about the future behaviour of an actor without any knowledge of an actor's earlier 'behaviour'. As the past and future are linked, previous experiences and expectations for the future are also linked. The past can be seen as a representation of the future. The future is a consequence of events in the past, just as expectations of an actor's behaviour are extrapolations based on knowledge of behaviour previously displayed by an actor.

In Norman's (1986) seven stages model (see section 3.8.3), the expectations are brought up to date by the evaluation of the interpreted perceptions. Expectations are formulated in relation to an actor's behaviour. On the basis of these expectations an action is executed now. Imagine the decision has been made to trust an actor. If the actor now behaves in a way that supports the trust and thus fulfils a person's goal, evaluation will result in the expectation being enhanced and the trust is enlarged. When an actor behaves in a way which damages the person's trust and therefore the person's goal is not fulfilled, this results in the expectation being diminished and thus trust is reduced.

When, for instance someone meets a new person (an actor B) he/she will try to classify this actor B and make some sort of character sketch (model) based on earlier experiences with other people. Earlier experiences with persons who behave in a way similar to the new actor B serve as the basis for expectations regarding the future behaviour by the new actor B. The referents' behaviour and the expectations of these referents' behaviour in certain situations are then used as a kind of mould to predict the new actor B's behaviour. As a person is confronted more and more with the new actor B's behaviour, the 'model' the person has of actor B will be refined and adapted until this actor B him/herself can serve as a mould for new people. The above means that trust is transposed from one actor to another. When a person then trusts a certain actor B and this person is later confronted with another actor C who displays many similarities with the familiar actor B, the knowledge will be transposed from the old and familiar actor B to the new actor C, and a quite similar image of trust is generated in relation to the new actor C. On the basis of this, a new hypothesis can be formulated:

Hypothesis 1:

> When a knowledge-based system is introduced, if there is familiarly with (knowledge about) technically high-quality apparatus, the trust in the knowledge-based system will be higher than in cases where this familiarity (knowledge) is not present.

To be able to research this hypothesis, two groups of persons have to be tested: the first group in which the subjects work with technically high-quality apparatus on a daily basis and the second group in which the subjects do not work with technically high-quality apparatus. A significant difference in the trust should be present between the two groups, where the first group has more trust in the knowledge-based system than the second group.

In this case, using the knowledge-based system ZKR (ZieKenhuis Roostering, a system for scheduling hospital staff) with both a technical and a non-technical group of users can test the above hypothesis. On an Intensive Care (IC) unit, a nurse has to rely on, and thus trust, electronic apparatus. This is a large part of the work on an IC unit. On a non-technical unit, the use of high-quality technical apparatus will be less prominent. One might assume that the trust in the ZKR knowledge-based system would be larger on an IC unit than on a non-technical unit.

3.8.3 Goals

In general, goals are the reason behind almost every human action. Depending on the goals formulated by a person, an action may or may not be executed. Goals are the basis of execution and evaluation in Norman's (1986) seven stages model (see figure 3-6). On the basis of a goal formulated by a person, an intention is contrived, and actions are specified and executed. Depending on the goal, the perception is interpreted and evaluated. An intention is formulated on the basis of a person's goal in his/her interaction with the outside world. These goals and intentions are the basis of the formulation of expectations that again can be used for the specification and the execution of the action.

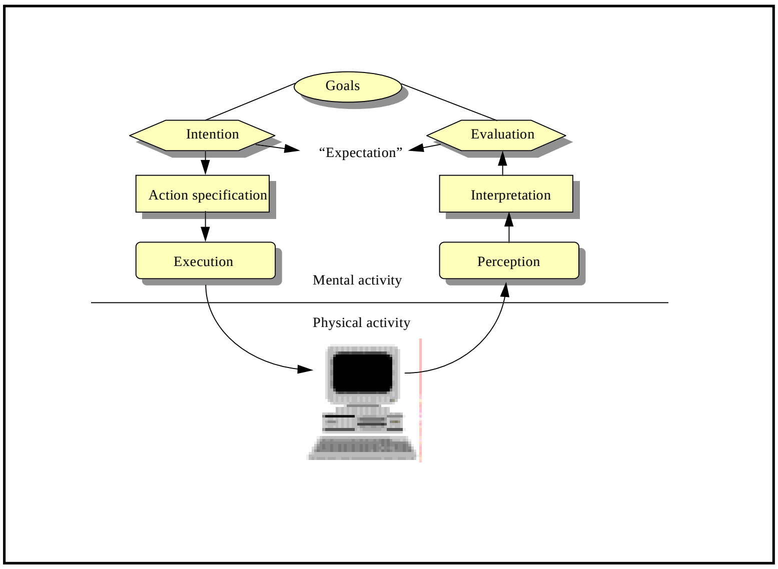

Figure 3-6: seven stages model (Norman)

The specification of the action could consist of the choice to trust or not to trust actor B in the physical world. Whether or not the choice to trust is made depends on a person's expectations about the actor's future behaviour with regard to fulfilling the goal or goals that the person has. When the actor does not support certain goals, the expectation that the goals will be achieved will be low, and therefore the trust that is to be measured will also be low. This hypothesis can be defined as follows:

Hypothesis 2:

> When a knowledge-based system will not support every important goal a user has, trust in the knowledge-based system will be lower than in cases where every goal of the user is supported.

In order to be able to research this hypothesis, two groups have to be tested: the first group, in which the user's goals are supported, and the second group, in which the user's goals are not supported. There has to be a significant difference in trust between the two groups: the first group must have greater trust in the knowledge-based system than the second group.
In this case, using the CASH knowledge-based system (Computer Aided Selection of Hearing Aids) can test the hypothesis in a commercial group and a non-commercial group. CASH is a system that selects hearing aids for people with a hearing disorder on the basis of the type of hearing deficiency and other personal data. The CASH knowledge-based system (Numan, 1991; 1994) was developed in a non-commercial environment (Groningen Academic Hospital) and does not take into account the commercial interests that a user can have. In addition to the aspect of the choosing a good hearing aid, the profit margin is also of importance in a commercial environment. This is not supported, therefore the users are not completely supported in both goals. Thus, there should be a significant difference between the commercial and non-commercial group of subjects. In the non-commercial environment, the trust in the knowledge-based system will be larger than that in the commercial environment.

3.9 Conclusion

The previous concepts are all interconnected. The previous discussion indicated clearly that trust between people is an important concept in social scientific research. It has also become clear that trust is not only of importance in the interaction between people, but also in the interaction with complex systems, such as complex knowledge-based systems. Furthermore, it was shown that trust is a concept that is essential in enabling people to cope with a large amount of complexity, and that trust influences different psychological constructs. In this chapter, an outline was given of what is meant by trust and which concepts are related to trust. In addition, the connection between trust and the psychological concepts related to trust has been dealt with. A definition has been given for every psychological construct used in this research. Furthermore, the hypotheses to be studied have been clarified.

In the following chapter, the method of research and the subject of research are dealt with.

4. Measurements and subject of research

4.1 Introduction

The aim of this chapter is to provide understanding about the measurements that are made in this research. Furthermore, the research subject is investigated and the used techniques of analysis are discussed. First, the measurements, which are implemented in order to measure trust, will be examined.
As discussed previously, trust is a construct that cannot be measured directly. This gives rise to various problems, namely: does trust exist and, if it does exist, how should the construct, which is not directly measurable, be measured? The second question will now be dealt with. How can something be measured which is not directly measurable? When something is not directly measurable it has to be measured indirectly.
The way of working in this research is analogous to the measurement of intelligence. It is not known what the construct of intelligence exactly is. However, in the field of measurement of intelligence, different systems have been developed in order to be able to make a measurement and to be able to make a statement on someone's intelligence in relation to the intelligence of other persons. If a person's intelligence has to be measured in IQ (Intelligence Quotient) in order to be able to make a statement on the measure of this person's intelligence, a good statement on the intelligence quotient could be made by using several intelligence measurement techniques.
Analogous to the previous example, this study will also try to measure trust in several ways, in which the different measures are an indication for the trust that a user has in a system.
Based on the theory in Chapter 3, four different measures are defined. Therefore, four different measurements will be executed: measures of behaviour from the logfile, reaction times, measures of behaviour from the observation protocols, and measurements with the aid of a questionnaire.

4.2 The measurements

4.2.1 Introduction

In this research, use was made of two knowledge-based systems (ZKR and CASH, both of which have been briefly discussed in Chapter 3). ZKR is discussed in more detail in Chapter 5 and CASH is discussed further in Chapter 6. These knowledge-based systems have been adapted for this research, so that the so-called logfiles could be used. Logfiles, in this research, are files which record what the user did and at which time. To be more specific, the recorded data consist of the user's input (what the user did), the point in time at which the user entered data in relation to the start-up of the program, and how much time has passed since the user's most recent input.
In the following part of this section, the measures that are obtained from the logfiles and the measure of the reaction times are discussed. Furthermore, an explanation is given of which measures can give an indication for the trust, which a person places in a system, and why these measures are an indication of trust.

When a user works with and trusts a system, he/she will, according to the definition of trust used (Chapter 3), expect the solutions generated by the system to correspond to the goal which the user had when he/she consulted the system. This means that the user does not wonder whether or not the solution that is generated by the system is correct, but expects this to be the case and therefore simply accepts that the solution will be correct. The essence of trust is, as has been explained in Chapter 3, the decision to hand over the control over an event to a person or a system (referred to as the 'actor' in this study). From this theme, different measures can be generated in cases where a system offers solutions to problems which a user wants to solve and where a system offers the possibility to give qualitative feedback on the correctness of the generated solution. The two preconditions of giving solutions and giving qualitative feedback to the user are features of most knowledge-based systems. The qualitative feedback is given on the basis of the dimensions generated with the knowledge acquisition that the solution has to meet. When the user is offered a solution by the system, the possibility to give feedback on the solution should to be given. This feedback then is linked to, for instance, the following dimensions: the measure of meeting the labour percentage, the measure of the spread of work per person, the measure of the complete spread of work, etc.
Thus, the two systems used (ZKR and CASH) have certain similarities. First, they both have a logfile in which the actions of users, such as the reaction times and the actions that are executed can be recorded. Furthermore, both systems offer the user a solution to a certain problem. Finally, both systems give qualitative feedback about the generated solution. From the preconditions, and based on the theory from Chapter 3, different measures can be generated. These are elaborated in the following sections.

We shall first deal with the solution-acceptance speed.

4.2.2 Measure 1: Solution-acceptance speed (TRREACT)

With the first measure, the central issue is the view that a person who trusts the system will accept the generated solution more easily than a user who does not have that trust. This measure is based on the principle of trust that the control over events is allocated to the person or system that has to be trusted. A user then does not have to check every possibility in order to investigate whether or not the system will generate the correct solution, and therefore the user is not burdened with all sorts of checks afterwards. A user who does not trust a system will tend to perform labour-intensive checks because he/she does not assign control to the system, but wants to compensate the lack of trust with the labour-intensive checks. In the example of the statistical package, a user would execute a lengthier check on the result of a statistical analysis when there is little trust in the program than when the user does trust the program. This means that the duration of examining the generated solution is an indication of the trust a person has in the system, where there is a negative correlation between the duration and the measure of trust. The first measure of behaviour then can be described as follows:

1. The higher the amount of user trust, the quicker the solution is accepted.

This measure makes it possible to apply an objective measure for measuring the size of trust. This measure will be called TRREACT from now on. It is a combination of TRust and REACTion speed.

The following measure of behaviour is subdivided into three different types of user behaviour. The final measure is a combination of these three types of behaviour.

4.2.3 Measure 2: measures of behaviour from logfile (TRBEHAV)

4.2.3.1 Introduction

The following measure is constructed out of measurements that are performed with the aid of a logfile. A logfile is a file that records every important action and command of the user. The recorded data are: the action which is executed, the total time that has elapsed from starting the program to the executed action, and the difference in time between one action and the previous action.

4.2.3.2 Measure 2a: number of studied solutions

With this measure it is assumed that a user is offered more than one solution. This means that the system (actor B) generates different solutions to the problem specified, and also shows these to the user. The user (person A) now has the possibility to choose one solution from those on offer. According to the definition of trust in Chapter 3, a user A will expect the knowledge-based system to offer good solutions. Furthermore, in Chapter 3 it was stated that a person who trusts an actor will hand over the control over (sub)tasks to this actor. In this case, then, person A assumes that actor B will execute the (sub)tasks correctly. Person A then allocates a part of the cognitive action to actor B; thus, person A relinquishes a part of the problem.

When user A trusts the actor B, the user A will not check whether or not every solution is correct, because user A trustingly expects that the solution will at least not be bad; after all, actor B will not offer bad solutions. A user who trusts actor B will accept any solution and will not look any further. It becomes much easier for a user when he/she trusts actor B and can simply make a choice from the solutions offered, without having to check every possibility. In a scheduling system, for example, different solutions could be generated and shown to the user when requested. This means that the number of solutions that is checked is an indication of the measure of trust which a person has in the scheduling system. This measure then can be described as follows:

2a. The larger a user's trust in the system, the fewer the number of solutions examined.

4.2.3.3 Measure 2b: changes of solutions

This measure assumes that a user who does not trust a system will try to 'improve' the solution. The principle of trust is that the user assumes it to be a fact that the system generates correct solutions. As discussed in Chapter 3, a user who trusts a system will not tinker with the solution provided by the system; he/she will assume the solution given is a good solution. An essential feature of trust is that person A assigns a (sub)task to actor B (in this case a knowledge-based system).

When a user starts adjusting the solutions, it would appear that the user does not trust the solutions that are generated by the system. Imagine that a scheduling program generates a schedule and offers the generated schedule to the user. The user who trusts a program will assume that the generated solution will meet the requirements regarding the quality of the solution. This measure then can be described as follows:

2b: The smaller the user's trust in a system, the more changes in a generated solution.

4.2.3.4 Measure 2c: qualitative information

This measure has, as a starting point, the fact that a user who trusts a system will make little or no use of the possibility of receiving qualitative information about the solution from the system.

As has become clear in Chapter 3 and the previous sections, a person A who trusts an actor B will not question the solutions provided in a certain (sub)task. The trusting person A will assume that actor B has acted as expected. This means that, when it is possible to ask for qualitative information, this action of asking for information is an indication of a lack of trust on the part of person A.
Checking the qualitative information about a solution is, in principle, an extra check on whether or not the system has generated the correct solution. Not looking at the qualitative information is a measure of the user's trust in the system. This measure then can be described as follows:

2c. The less trust a user has in the system, the more frequently qualitative information is requested.

4.2.3.5 Conclusion

The measures discussed are possibilities to have an objective measure for measuring the size of trust. Summarising, they consist of the following measures:
2a. number of studied solutions;
2b. number of changes of the solution;
2c. number of requests for qualitative information on solutions.

The measures 2a, 2b, and 2c are combined into one measure that, from now on, is called TRBEHAV. The name TRBEHAV for this measure is a combination of TRust and BEHAViour. The measure is a sum of the scores on 2a to 2c. The summation is done on the rough scores, without weighting them, because no distinction can be made between the importance of the separate scores. Every separate score is a separate observation. After this summation, this figure is multiplied by -1, because a high score points to a low score on trust. After the multiplication, a high score on the scale is a high score on trust.
In the following section, other measures of behaviour of trust, which are based on observations of users with computer systems, are dealt with.

4.2.4 Measure 3: measures of behaviour from the analysis of protocols (TRCATEGO)

4.2.4.1 Introduction

A technique, which is used in much psychological research, is the so-called observation technique. This means that people are observed while they execute a task and that the observation is recorded in some way. The result of recording observations is called the observation protocol. Subsequently, the observation protocol that has been created can be analysed using various techniques.
This research also made use of an observation technique, namely, the so-called 'think-aloud protocols' (Ericsson & Simon, 1984; Breuker & Wielinga, 1986; Hart, 1989; Kidd, 1986; Lueke et al., 1987; Reitman-Olsen & Reuter, 1987; Wielinga & Breuker, 1988). Think-aloud protocols are, as the term indicates, protocols of people who execute a task and express aloud what is on his/her mind during the execution of the task. The statements of the subjects are recorded on tape and transcribed later on. The transcription is analysed afterwards, which can happen in various ways. In this case, the choice was made for a categorization system. This means that categories, in which a remark can be placed, are formulated in advance. The scores on the various categories are finally combined into one measure.

In the definition and the model of trust, it became clear which aspects of a computer and a program influence a user's trust; these are: the computer, the program, the experiences with the program and the computer, expectations, looking-for-help behaviour, trial-and-error behaviour, behaviour based on former experiences and uncertainty. They will be dealt with hereafter.

4.2.4.2 The observation measures

Earlier experiences with a program influence the trust in that program. As was stated in Chapter 3, familiarity with a program is an essential precondition for building up trust in a program. Positive experiences will have a positive effect on trust and negative experiences will have a negative influence on trust. This means that the number of positive and negative statements about a program, in relation to earlier experiences with the program, indicate an increase and a decrease of trust in the program, respectively. This results in the following two categories:

1. Negative remarks about previous experiences with the program: Frequency
2. Positive remarks about previous experiences with the program: Frequency

Analogous to this reasoning, previous experiences with a computer will have influence on the trust in that program. Users often find it difficult to indicate the difference between a program and a computer. Positive experiences will have a positive effect on trust and negative experiences will have a negative influence on trust. This means that positive and negative remarks about the computer, in relation to previous experiences, indicate an increase and a decrease of trust in the program, respectively. This results in the following two categories:

3. Negative remarks about previous experiences with a computer: Frequency
4. Positive remarks about previous experiences with a computer: Frequency

Expectations of a program's future behaviour are indications of a person's trust in a program. Expectations are the basis of trust that a person has in a computer program. The definition in Chapter 3 illustrates that expectations are the basis of trust.

Summarising, the positive and negative remarks on the expectations of a program indicate respectively an increase or a decrease of trust in the program. When someone works with a program and makes a remark like: "It probably won't get this" this indicates a lack of trust.
Analogously, it can be said that positive expectations indicate a positive trust. This results in a score in the following two categories:

5. Negative expectations of a program: Frequency
6. Positive expectations of a program: Frequency

Expectations of future behaviour of a computer are indications of a person's trust in a program. As mentioned previously, a user often finds it difficult to indicate the difference between a program and a computer. This means that positive and negative remarks on expectations of a computer indicate an increase and a decrease of the trust, respectively. When someone works with a computer and makes a remark like: "I think this thing will get stuck again", it indicates a lack of trust. Analogously, it can be said that positive expectations indicate a positive trust. This results in the following two categories:

7. Negative expectations of a computer: Frequency
8. Positive expectations of a computer: Frequency

The appearance of the results expected from a program's behaviour increase a person's trust in a program. This means that expected and unexpected results indicate respectively an increase and a decrease of trust in the program. A remark like, for example: "What has it been doing now?" is a remark on unexpected results of actions with the program. This results in the following two categories:

9. Unexpected results of a program: Frequency
10. Expected results of a program: Frequency

The results expected from a computer's behaviour increase a person's trust in a program. This means that expected and unexpected results from a computer indicate respectively an increase and a decrease of trust in the program. A remark could be, for example: "Where has this computer left those files this time?". This results in the following two categories.

11. Unexpected results from a computer: Frequency
12. Expected results from a computer: Frequency

One of the ways of building up trust is by requesting help from the Help system or from a person. The generation of this type of trust was explained in Chapter 3. This sort of trust is type 1. Asking for help is an indication of a person's lack of trust in a program. As has been stated previously, in the discussion of the model of trust, asking for help is an attempt to remove a lack of trust. This results in the following two categories:

13. Asking for help from a person: Frequency
14. Asking help from the Help system of a program: Frequency

The use of a program without thinking is an indication of a complete trust in the program. Furthermore trial-and-error behaviour is an indication of a certain level of trust. In Chapter 3, an explanation was given about this second type of trust in the model of trust.

Chapter 3 also explained that a person (person A) who has trust in something (actor B) will have contact with this actor in such a way that he/she anticipates only positive actions being performed by actor B. This means that a person A no longer has to think about the possible negative consequences of interacting with an actor B; consequently, this person A can act without a certain degree of cautiousness. When these two types of actions are undertaken, this means that the frequency with which these actions are undertaken is a positive measure for the degree of trust. This results in the following two categories:

15.	Giving commands without thinking:	Frequency
16.	Trying out parts of the program:	Frequency

4.2.4.3 Conclusion

In the previous section, various observation categories are distinguished upon which a measure of trust can be based. This measure will subsequently be referred to as TRCATEGO.

The name TRCATEGO is a combination of TRust and CATEGOry system. TRCATEGO is constructed by adding or subtracting every score. This means that negative remarks are subtracted from the total and positive remarks are added to it. The following division was made:

1. Negative remarks on earlier experiences with the program:	Negative
2. Positive remarks on earlier experiences with the program:	Positive
3. Negative remarks on earlier experiences with a computer:	Negative
4. Positive remarks on earlier experiences with a computer:	Positive
5. Negative expectations from a program:	Negative
6. Positive expectations from a program:	Positive
7. Negative expectations from a computer:	Negative
8. Positive expectations from a computer:	Positive
9. Unexpected results from a program:	Negative
10.Expected results from a program:	Positive
11.Unexpected results from a computer:	Negative
12.Expected results from a computer:	Positive
13.Asking for help from a person:	Negative
14.Asking for help from the Help system of a program:	Negative
15.Giving commands without thinking:	Positive
16.Trying out parts of the program:	Positive

A negative overall score thus means that the person has a more negative attitude than a positive one, while the number says something about the size of the negative attitude.

As was mentioned, a fourth way of measuring trust is having the user fill in a questionnaire. In the following section, the questionnaire will be dealt with briefly.

4.2.5 Measure 4: The questionnaire (TRQUEST)

4.2.5.1 Trust

Three important aspects of the definition of trust are considered.

Firstly: Trust is based on past experiences.

Secondly: Trust is always oriented towards the future and can be considered to be an expectation.
Thirdly: Trust always has a specific actor.

These three elements should be tested with the questionnaire.
Thus, there are two parts of the questionnaire (past experiences and expectations) both of which refer to the same actor. In this specific case, a second condition is applicable, namely the fact that this particular test is a test on computer programs. The words 'computer' and 'program' already suggest that two aspects should be tested: the computer and the program. As stated previously, a user often does not distinguish between a computer and a program. For a user, these two aspects are strongly intertwined. The previous considerations result in the following matrix:

	Computer	Program
Previous experience	**I**	**II**
Future expectations	**III**	**IV**

Table 4-1: elements of questionnaire

The questionnaire has to address every cell in the above matrix. For details of the questions themselves, see Appendix 1. Cell I in this matrix indicates that questions are asked on previous experiences with the computer, while cell II covers questions which are linked to previous experiences with the program. Cell III deals with the expectations of using computers, while cell IV covers questions dealing with expectations of the program.

4.2.5.2 Conclusion

In the previous section, the questionnaire was discussed very briefly. This measure will subsequently be referred to as TRQUEST. This name is a combination of TRust and QUESTionnaire.

4.2.6 Other measurements

In addition to the measurements of trust, some other measurements are also necessary, as the previous section indicated. One of those measurements is the measurement of anxiety regarding computers. The computer anxiety questionnaire was based on an existing questionnaire: CAIN (Computer Anxiety Index) by Maurer (1983) (see Maurer, 1984; 1994). In addition to the anxiety questionnaire, a computer attitude questionnaire was administered to every subject. The attitude questionnaire was also based on an existing questionnaire: CAS (Computer Attitude Scale) by Nickel and Pinto (1986). In Chapters 5 and 6, the validity of the anxiety and attitude questionnaire is discussed. Besides the previous data, the extent of computer expertise is also measured by means of a questionnaire including the following items:

1. How long have you worked with the program?

2. How many hours a week do you work with this program on average?
3. How long have you worked with a computer?
4. How many hours a week do you work with a computer on average?
5. Which of the following systems have you used at some time?
6. Which of the following machines have you worked with at some time?
7. Which of the following programs have you worked with at some time?

Furthermore, the following data were also measured:
1. Gender
2. Age
3. Education: this measurement consists of three levels: "Secondary", "Higher" and "University".

The following section deals with what the four measures of trust exactly measure.

4.3 What do the four measures measure?

In the previous sections, the four different measures that have to measure trust were examined. In the definition of trust in the previous chapter, it was clear that trust is based on the expectations that a person has of an actor. These expectations are based on certain knowledge which is acquired during the interaction with the actor and is not completely proven to be absolutely reliable to the trusting person. It can be stated that, on the basis of the acquired knowledge, there is a certain 'trust attitude' which a person has regarding the actor who is to be trusted. This basic attitude is measured with the trust questionnaire (TRQUEST) in this study. Moreover, trust is a construct that is not static but has a dynamic character. This implies that, during the interaction with the actor, the trusting person will experience fluctuations in the trust placed in the actor. These fluctuations are recorded by means of the TRCATEGO scale. In the previous two measurements, the trusting choice does not yet really emerge. Trust is a mental action that emerges during the interaction with the actor. In general, interacting with a complex system means that a user will allocate certain parts of the task to the system. The extent to which a user assigns certain parts of a task to the system and does not check whether the system is doing the correct things gives, in general, a good opportunity to measure the mental trust. TRBEHAV is a measure that measures this trusting action. When a person does not check whether or not a solution is adequate, this can be called a trusting action. Finally, trust also has consequences for other aspects in the interaction with a system. When a person trusts an actor, this will speed up the interaction process. This means that the reaction time of a trusting person is quicker than that of a non-trusting person. This measurement is called TRREACT. Thus, the measurements cover the following parts:
1. Basic trust attitudes (TRQUEST)
2. Trust fluctuations during the interaction (TRCATEGO)
3. Trusting action (TRBEHAV)
4. Consequences of trust (TRREACT)

The four scales of trust should each measure a part of trust, and there will be some overlapping. In fact, there is a matter of converging evidence. This means that the scales have to cover the construct of trust collectively, and thus, taken altogether, they are an indication of the measure of trust. In the following section, the subject of research is discussed.

4.4 Subject of research

As explained in the hypotheses in Chapter 3, two different types of groups are required to carry out the research. The first group is the technical/non-technical group and the second group is the commercial/non-commercial group.

The technical/non-technical group is divided into its two component groups. In one subgroup, there is not a large familiarity with technique (they are 'non-technical') and in the other subgroup, there is some technical background and so there is some familiarity with technique ('technical').

With the group commercial/non-commercial, a division was also made. In one subgroup, not every goal is supported ('commercial'), while in the other subgroup, every goal is supported ('non-commercial'). The following table displays a graphical representation of these groups.

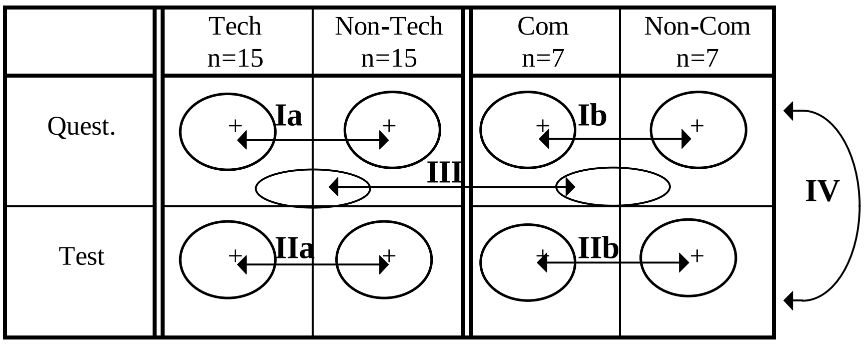

Table 4-2: subjects of research

The arrows I to IV represent the following:

I. Here, groups are compared on the basis of the questionnaire; the independent variable consists of the Technical vs. Non-technical and the Commercial vs. Non-commercial categories. The dependent variable is the score on the questionnaire.

II. Here, the groups are compared on the basis of the test data (observations and behaviour regarding the solutions); the independent variable consists of the Technical vs. Non-technical and the Commercial vs. Non-commercial categories. The dependent variable is the scores on the tests[2].

III. Here, the groups are compared on the basis of the dependent variables of both the test score and the questionnaire score; the independent variables are the use of a planning system vs. a diagnosis system.

[2] On average, arrows I and II will be related to one another later on, but they are part of the same hypothesis.

IV. This arrow is meant to indicate that the two measures (questionnaire and test data) are connected, and both have to produce an indication of trust.

4.4.1 Expectations

The expectations of the results with each arrow in Table 4-2 are mentioned below:

I. The expectation with arrow I is that, among the subjects in the technical group who make use of ZKR, the trust in ZKR, as measured by the questionnaire, will be larger than that among the non-technical group, because of the 'technical' character of the environment. Another expectation with arrow I is that, among the subjects in the Non-commercial group who make use of CASH, the trust, as measured in the questionnaire, will be larger than that among the Commercial group, because the program does not bear in mind the users' commercial motives.

II. The expectation with arrow II is that, among the subjects in the technical group who make use of ZKR, the trust in this program, as measured by the test results, will be larger than that among the Non-technical group, because of the 'technical' character of the work environment. Another expectation with arrow II is that, among the subjects in the Non-commercial group who make use of CASH, the trust in CASH, as measured by the test results, will be larger than that among the Commercial group, because the program does not bear in mind the users' commercial motives (goals).

III. The expectations with the use of a planning system vs. a diagnosis system will be that, in a diagnosis system, trust will be either high or low, while, with a planning-system, the trust has a more continuous character. This is because the risk with a diagnosis system is larger than that with a planning system.

IV. The expectation with arrow IV is that the two types of data will be closely connected. When this is not the case, the two types of data do not measure the same construct.

The above indicates that the groups are selected on two different group characteristics, namely: supporting every goal and the familiarity of the users with computer usage. The handling of both experiments is the same for the two subgroups, namely, the use of a diagnosis system or a planning system. This results in the groups having to be comparable and only differing, in principle, in the classification variable. Because both groups receive the same comparable treatment, a difference between the two groups has to be ascribed to the classification variable, if a difference is found.
In the following section, the type of research is discussed.

4.5 Type of research

In this research, the groups that are opposed (Technical vs. Non-technical and Commercial vs. Non-commercial) have to be comparable and may only differ on the classification variable. Because both groups receive the same treatment, a difference between the groups has to be ascribed to the classification variable, if a difference is found between these groups. This design most resembles the so-called post-test only non-equivalent peer control group design (Neale & Liebert, 1995). In this research, an attempt was made to indicate a causal connection between the independent variable (membership of a group) and the dependent variable (trust). In order to be able to do this, three different criteria have to be met (Baarda & De Goede, 1990, p.102):

1. A co-variation or a statistical correlation between the independent and the dependent variable has to be present.
2. The variable, which is regarded as the independent variable, has to precede the dependent variable in time.
3. It has to be proved whether the correlation between the independent and the dependent variable is false or not.

After the execution of the tests, the first assumption can be checked. The second assumption is indeed clearly the case in the previous subject of research. The groups already existed before the treatment was given to the two groups. The third criterion is met by comparing two related groups on different important characteristics, such as age, education, knowledge of computers, etc. Furthermore, the same occupational group with comparable education has been drawn from for both the technical and non-technical group. Therefore, it can be expected that the groups will score similarly on almost every important dimension with the exception of the technical and the non-technical environment. Therefore, it can be reasonably assumed that any differences found between the groups should to be ascribed to the influence of the technical or non-technical environment, and not to a third intervening variable. In the following two chapters (chapters 5 and 6), these three aspects of the causal relation between the independent and dependent variables are gone into in more depth.

4.6 Case studies

This research was not directed towards case research. In the field of business administration, case research is executed regularly. Therefore, a section is devoted here to the reasons why case research has not been applied here. The main reason is that case research brings problems with the generality of the results. In contrast to case research, statistical and experimental research provide possibilities to check moderator factors in order to keep their influence as minimal as possible. Thus, a measurement as accurate as is possible is obtained of the subject in which the researcher is interested. With statistical research, it is important to keep the number of objects of research as large as possible so that any moderator factors are averaged out as much as possible. By executing an experiment, an attempt is made to control the moderator factors as much as possible so that any effect that may appear can be ascribed to the treatment.

Some attempts have been made to increase the generality of a case study. First, more case studies on the same subject can be executed. Unfortunately, the question still remains as to whether the research can be controlled sufficiently for the various cases, because almost every case research is characterised by a more qualitative approach. A second solution that is proposed is to have several researchers doing several cases. Nevertheless, the problems of the qualitative approach remains. A third solution that is proposed is to either describe a typical case which represents a class of cases or to select a 'deviant' case which can falsify a theory. However, the problem remains that coincidence factors and moderator variables are difficult to control in an experiment; in a case study this is almost impossible to eliminate.

In principle, the exemplary evidence of the case study only says something about the object that is being studied and nothing about the total population. However, a case study can be used in an explorative and descriptive fashion in order to be able to generate a hypothesis, for example. These generated hypotheses could be tested with an experimental design. Unfortunately, case study cannot be used for testing and/or explaining.

Because of the reasons mentioned, no use was made of case studies here. The aim of this research was to explain and test. The research has approached the empirical experimental situation as much as possible.

4.7 Conclusion

In this chapter, the measurements that are used to measure trust were examined. Four different measures are used, namely: measures of behaviour from the logfile (TRBEHAV), reaction times (TRREACT), measures of behaviour from observation protocols (TRCATEGO), and measures applied by means of a questionnaire (TRQUEST). With the aid of these measures, the construct of trust becomes measurable. Therefore, it becomes possible to obtain understanding of the construct of trust. The four measures each measure a part of trust and give an indication of trust. Basic trust attitudes are measured by TRQUEST. Trust fluctuations during the interaction are measured by TRCATEGO. Trusting actions can be measured by TRBEHAV, while the consequences of trust can be measured by TRREACT. It appears that these four measures can be used as good indicators of trust. Furthermore, it appears that the design most resembles a so-called post-test only non-equivalent peer control group design.

5. Test I (ZKR)

5.1 Method of Test I

5.1.1 Introduction to Test I

In Chapter 3, the two hypotheses that are to be tested were discussed. This chapter describes the way in which the measure of familiarity influenced the measure of trust. The suggestion was made to test this influence of familiarity on the basis of two different groups, namely a technical and a non-technical one. In Test I, the first hypothesis is tested, namely:

> When a knowledge-based system is introduced, if there is some familiarity with (knowledge about) technically high-quality equipment, the trust in the knowledge-based system will be greater than in cases when this familiarity (knowledge) is not present.

The previous hypothesis can be transformed into the following testable hypothesis:

> A significant difference in the scores between the non-technical group and the technical group is present on the four trust scales, where the technical group has proportionally more trust in the program than the non-technical group.

As mentioned, two groups of users are necessary to be able to test this hypothesis: the first group, whose members are not familiar with technical equipment of this type, and the second group whose members do have a certain degree of familiarity with technical equipment. In this case, the ZKR knowledge-based system (*ZieKenhuis Roostersysteem*, Hospital Scheduling System) can be used as stimulus material, where the first group consists of non-technical users and the second group of technical users. First, the background and working of the ZKR program ZKR will be further explained in the following section.

5.1.2 The stimulus material of Test I (the ZKR program)

In this test, the subjects were presented with a program that generates schedules, interactively with the user, for the hospital staff (mainly nurses). The program used is a schedule planning system (ZKR). Planning schedules is an extensive area of research in which much research has been done on the so-called schedule-deciding-supporting systems (Mietus, 1994; Jorna et al., 1996). Before the system that was used is described, an outline of the background of the ZKR system and the theoretical background of scheduling will be given.

5.1.2.1 Background of ZKR

ZKR is a schedule-planning system which was developed by the DISKUS Foundation as a non-commercial prototype (Numan, 1992a; 1992b; 1994b). Later on, the ZKR prototype was further developed into a commercial system by the *Instituut voor KennisSystemen*, IKS (Institute for Knowledge-Based Systems). The IKS is currently

busy developing the program further. At this moment, the commercial version is working in various hospitals.
In this research, the commercial version of ZKR is used, having been lent by the IKS for this research. The version of ZKR used here is the 1.2 version. This version works on a PC running MS-DOS 6.0 and at least MS-WINDOWS 3.1. Furthermore, a 386-processor machine with at least 8 Megabyte internal memory is required. Because ZKR 1.2 works with a code of shifts that is based on characters and also on colours, a colour monitor is needed. ZKR 1.2 has become a product that can be used by new users after half-an-hour of instruction. The schedule generator of ZKR 1.2 ensures that both experienced schedule makers and inexperienced schedule makers are able to use the product capably. The inexperienced schedule makers can generate schedules which are of a good quality, while the experienced schedule makers can also generate schedules for obtaining ideas for solutions to subtasks, for example, or for gaining time with a shorter scheduling task. In the following subsections, the working of ZKR 1.2 is discussed further.

5.1.2.2 Theoretical background to staff planning or shift scheduling

The terms 'staff planning' or 'shift scheduling' refer to the following, according to Jorna et al. (1996):

> *"[...] harmonising limitations on the area of blocks of time, a large number of shifts, a non-homogeneous group of employees with limited availability, eventually on different locations in order to execute the different tasks."*

With the execution of staff planning or shift scheduling, six different stages are distinguished in the subtasks (Jorna et al., 1996), namely: administrating, counting, harmonising, adapting, valuing and negotiating.
Administrating mainly consists of recording and consulting all sorts of data. These data are of essential importance in making a schedule. The administration data, such as contract data and personal details, are combined with the data of the shifts etc., by means of which, in fact, the schedule is determined. Administrating is an essential part of the scheduling process. When scheduling with the aid of a scheduling system, administrating is an action that takes care that the database is filled. The database is, as has been discussed in Chapter 2, an important part of many knowledge-based systems. The database contains the many data that a system needs to execute its reasoning process.
Counting is, in principle, counting scheduled shifts and comparing the totals to the desired totals. Counting is an extraordinarily important part of the scheduling process because, with most scheduling problems, it is necessary to know how much is needed from a certain object for which time or place. Therefore, counting is a labour-intensive element in many scheduling problems. Many of the users of ZKR are generally very pleased with the runtime counting of the system purely because counting is very labour-intensive. A scheduling system can relieve this task considerably.
Harmonising is, in fact, what scheduling mainly refers to. It is the harmonising of, for instance, the people who have to be scheduled on certain times and/or shifts. In harmonising, much is demanded of the scheduler's cognitive system, because the schedule maker has to link up all sorts of data, often on the basis of many rules and restrictions. When scheduling staff, it is often of importance that a fixed number of persons are available each day, who collectively possess at least the minimum of

quality/experience. Furthermore, it is often of importance that there is a certain continuity in running certain shifts for those involved (nurses and patients, for example) and a certain length of time must, of course, be worked by those scheduled. In ZKR, the module that takes care of harmonizing is called the generator. This generator produces the schedules that are on offer, and these are then looked at by the evaluator and controller to see whether they are suitable. The evaluator appraises the schedule on the extent to which the schedule meets the specified goal functions. These goal functions are, in fact, a sort of objective which a schedule maker has, something like a proportional division of shifts per person in time. Goal functions are yes/no rules, but have a continuous character. The controller checks the schedule against fringe conditions introduced by the schedule maker. Fringe conditions are conditions that the schedule has to meet. Fringe conditions have a yes/no character; an example is the fringe condition that a person cannot work for more than ten days in a row. Generating schedule proposals with the aid of the generator can enormously relieve and speed up the task of making a schedule.

When a schedule is generated, it rarely happens that this schedule is run just as it is formulated. It regularly happens that a person falls ill or is prevented from working a shift. At that point, the existing schedule has to be adapted.
Adapting is mainly a subtask that is performed on an existing schedule and in which changes are implemented while trying not to violate the fringe conditions.
During the whole process of scheduling, the schedule maker is constantly busy considering the (potential) value of a future schedule.
Valuing is a subtask that is mostly executed after a schedule is formulated. Here examination is carried out as to whether or not the quality of the schedule can be increased, thus, whether or not the goal functions can be increased. Valuing is, in fact, checking whether or not the schedule still meets the fringe conditions. In ZKR, this is done by the controller, as explained in the following section.
The final subtask that is described consists of *negotiating*; here, the planner tries to tally the schedule. This negotiating is particularly important when a person has to be scheduled and that does not completely fit within the fringe conditions, or when another shift has been scheduled for the new shift which has to be scheduled. Here, a schedule maker should try to negotiate with the person who has to be scheduled.
The above subtasks are not always completed in the order given. One can even state that this is often not the case. In the following section, the working of the ZKR 1.2 program is discussed and the parts of ZKR will be related to the characteristics of the scheduling process.

5.1.2.3 The ZKR 1.2 program

The ZKR 1.2 program is a knowledge-based system that supports the user in making staff schedules for nurses in hospitals. This means the system makes various schedule proposals, in conjunction with the user, for the different shifts which have to be scheduled. The system supports the scheduling subtasks, mentioned in the section on the theoretical background of scheduling. The only scheduling subtask that the version used here does not yet support is negotiating. In the following subsections, each of the scheduling subtasks is discussed and an outline is given of how ZKR 1.2 supports this task.

5.1.2.3.1 Administrating

Administrating is oriented towards recording and consulting all sorts of data. In this case, this mainly consists of recording and looking at the data of the persons who have to be scheduled. These data can mostly be divided into personal details, the function data and the contract data. When a schedule has to be generated, these data have to be taken into consideration. Here, contract data in particular play an important part. The contract data cover areas such as the shifts a person is allowed to work. A member of staff who, for instance, does not have a contract to work night shifts cannot be deployed. Another example of the administration is ensuring that there is a correct staff file. Here, adding new persons and removing others are means to ensure that the basis data of a certain ward are correct. The ZKR 1.2 program keeps a record of the number of shifts worked, the numbers of days of leave taken, etc. These data do not have to be explicitly kept up to date by the user. The user, however, does have to record which personal details, function data and contract data have changed. The user can do this in an intuitive way. The principle is as follows: the user double clicks on the person whose data have to be adjusted. At that moment, the following window appears on the screen.

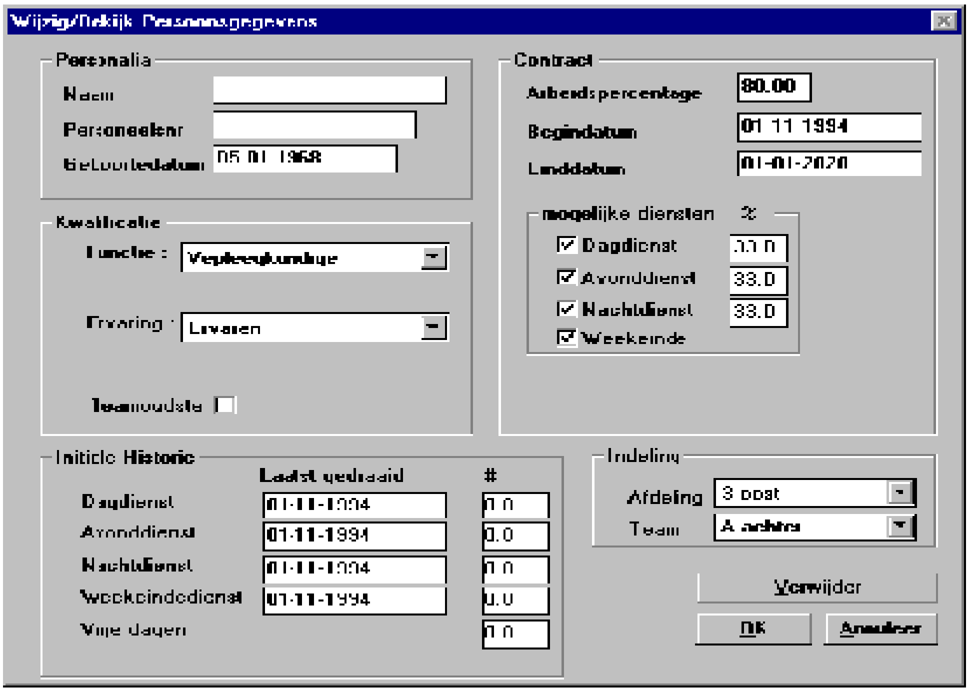

Figure 5-1: personal data entry window

In the above screen, the personal data can be entered. Furthermore, the file of the persons who have to be scheduled can be maintained by adding and removing persons. A specific part of the ZKR 1.2 program is the adjustment of the fringe conditions of the schedule and the schedule goal functions. Before there can be any mention of scheduling, the schedule maker has to know clearly which quality of staff has to be present on the workfloor for every shift, for each day. Every schedule maker also has to know how many people have to be present on the workfloor per shift, each day. With these rules, a distinction can be made between rules that have a yes/no character and rules that have the character of a scale. With ZKR 1.2, a distinction is also made between the fringe conditions and the goal functions. The fringe conditions are adjusted on the basis of yes/no options. Fringe conditions are conditions which are on or off. When the user has switched the fringe conditions on or off, a schedule can be checked with the aid of a controller. The controller then examines which fringe conditions are violated and indicates where problems with the fringe conditions arise in the schedule. The goal functions are of another order than the fringe conditions. The goal functions have the character of functions that do not have absolute values but are measured on a scale of values. An example of a goal function is recognising the member of staff's special wishes as much as possible, in a way that every specified fringe condition is met. It will be clear that not every wish can be honoured when many wishes have been submitted. This means that an attempt is made to honour the wishes as much as possible. Within ZKR 1.2, these goal functions and fringe conditions can be adjusted by the schedule maker, after which ZKR 1.2 takes into consideration the values adjusted by the user.

5.1.2.3.2 Counting

A very important part of scheduling is counting the scheduled shifts. Counting is generally done in two ways. The first way consists of counting the number of shifts per day. This means that, every day, an examination is made of how many day shifts, night shifts etc. are scheduled. This is of great importance for the final schedule, because, in general, it is known for every daily shift how many shifts have to be worked. Therefore, a schedule maker can see whether he/she is finished with scheduling a certain shift on a certain day. The second way of counting is counting on a personal level. This means that, for each person, a check is made of how many shifts this person has been assigned at that moment. The two counts are also referred to as vertical and horizontal counting. These terms have their origin in the matrix structure frequently used in planning, where the dates are presented horizontally and the names of the staff members vertically. The units that have to be scheduled are placed on the crossing. See also the following figure representing ZKR 1.2

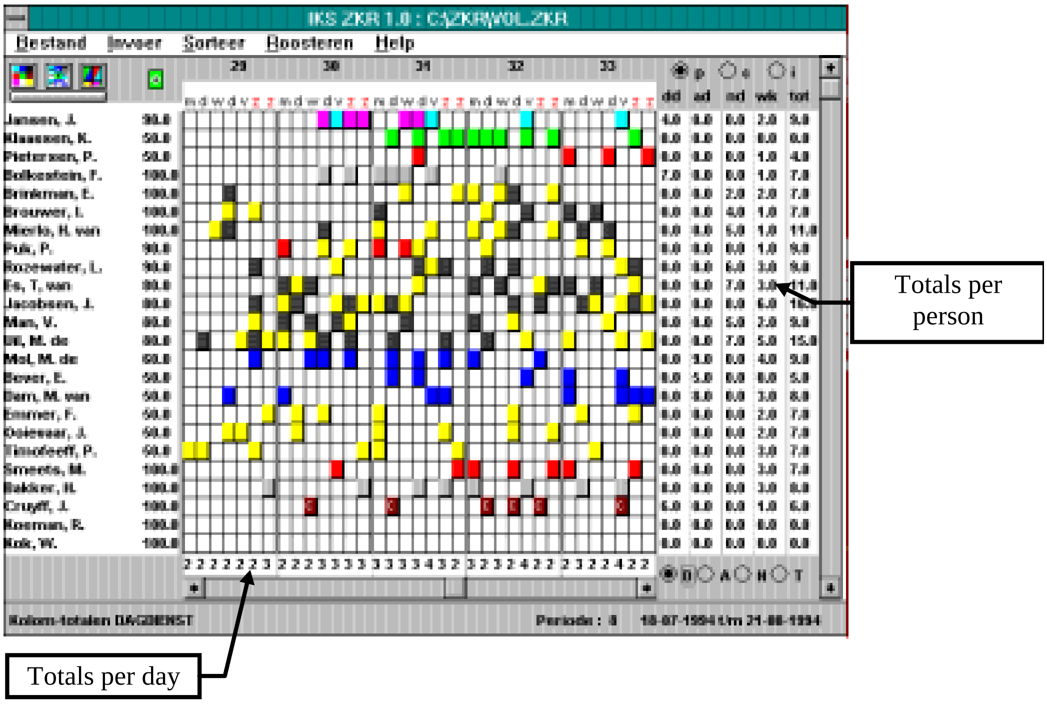

Figure 5-2: schedule matrix window

It is extremely easy to count using ZKR, because this entity is supported by the system. Every action on the schedule is examined and, where necessary, counting is adjusted in runtime. This means that, when specifying a shift, the counts are immediately updated in order to meet the new situation.

5.1.2.3.3 Harmonising

Harmonising is, in fact, scheduling. Mostly this means attuning a person to a shift at a certain time. When the user has adjusted each fringe condition and goal function, a schedule can, in principle, be generated. The program makes different schedule proposals. These proposals can be examined by the user and may be accepted or not. Here, an important aspect of ZKR 1.2 emerges, namely, the fact that the program supports, and does not decide, what the user can do. The user always has control over the program. The following figure illustrates how ZKR 1.2 offers the proposals to the user.

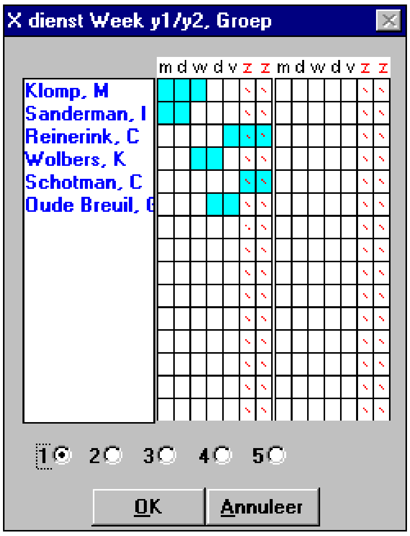

Figure 5-3: schedule proposal window

The user now can click on the selection buttons and look at the five different schedule proposals.

5.1.2.3.4 Valuing

When the user has instructed ZKR 1.2 to create a schedule, ZKR 1.2 will return with five proposals. If there is a good proposal it can be accepted by clicking on the OK button. If there is not a proposal to the user's liking, the Cancel button can be used to cancel the choice. At this point, the user has to consider what is a good and what is a bad schedule proposal. This is the aspect of valuing the created schedule. When a proposal is accepted, it can be examined, with the aid of ZKR 1.2, whether or not the schedule meets the fringe conditions. When a person who, for instance, is not allowed to work night shifts is scheduled to do so, a fringe condition is violated and the user, having given the Control command, gets the message that a fringe condition is violated. In practice, valuing a schedule is an essential part of the scheduling process. With the introduction of a scheduling program, which generates schedules, a clear move from harmonising to valuing can be noticed. The program generates schedules, while the user has to decide which schedule is best for a certain situation. Of course, schedules can also be manually generated in ZKR 1.2. Often the introduction of schedules will also involve adapting a schedule. The following subsection deals with this in more depth.

5.1.2.3.5 Adapting

Adapting is an essential part of scheduling. This adapting often happens at the point when someone falls ill. This means that a person cannot work a certain shift and thus that replacement has to be found. At this moment, the user often enters some sort of waterfall of cause and effect, a kind of domino effect. When, for instance, the head nurse is ill, a very experienced nurse should take over the tasks and thus also the shifts. Often the nurse standing in for the head nurse will work shifts too, which then have to be taken over by another experienced nurse. One does not need much power of imagination to understand that this can have considerable consequences for a long period of time. The process of adapting is very well supported by ZKR 1.2. In fact, ZKR 1.2 is constructed in such a way that the screen has become a kind of electronic planning board. This means that the users of the planning board, which most schedule makers are, can make use of the knowledge they have from 'real life' when working with ZKR 1.2. If a schedule maker wanted to assign a person to a shift on a certain day, when planning with a planning board, he/she would pick a block from a box and place this on the intersection in the matrix between the person and the day in question. With ZKR 1.2, this happens in an analogous way. See the following figure.

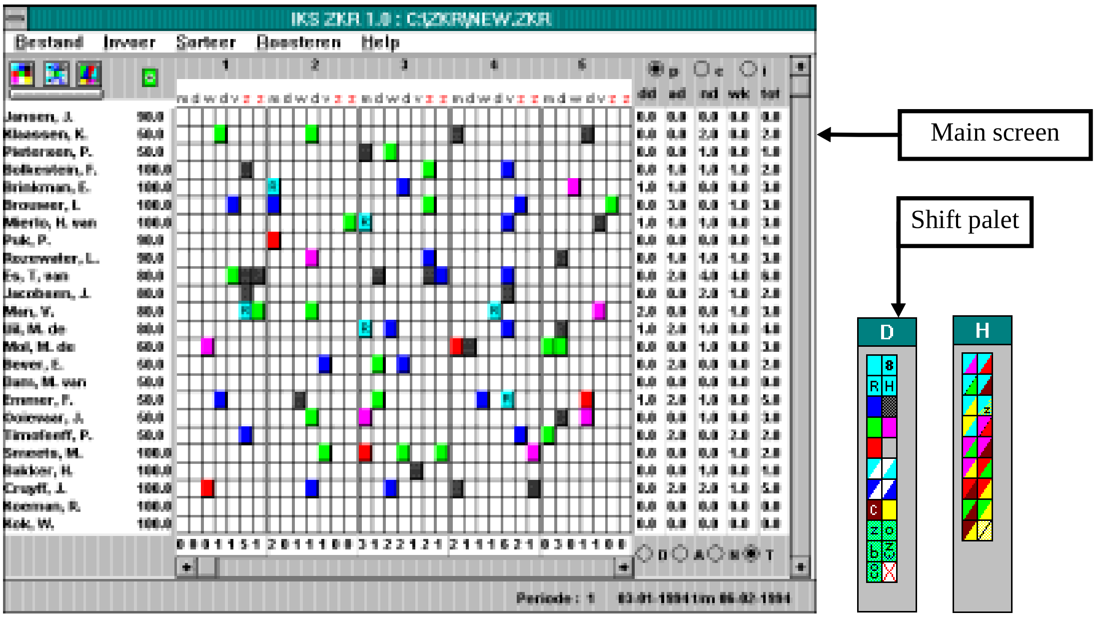

Figure 5-4: shift boxes

The schedule maker first has to take a shift from a box. In ZKR 1.2, this happens by opening a window with the shifts in it. After that, he/she has to select a shift that has to be scheduled using the mouse. This selecting is, in fact, the equivalent of picking a block out of a box. When this shift is selected it can be 'placed' on an intersection between a day and a person. At that moment thus, the person receives a shift on that particular day. When a shift is selected the analogy goes a bit wrong because, at that moment, this shift can be 'placed' as often as the user wants to without selecting it again. Most aspects of the functionality of ZKR 1.2 were discussed in the previous subsections. On the basis of these functionalities, some assignments, which had to be executed with the program, were formulated for the subjects. In the following section, the assignments, which the subjects received, are dealt with.

5.1.3 The user assignments

The test with ZKR, then, is executed with two groups of persons who differ, in principle, only in the degree of familiarity with technically high-quality products. One group is familiar with technically high-quality products and is called the technical group, while the second group is not so familiar with these products. The second group therefore is called the non-technical group. The program with which the test is performed is a program that can generate schedules; here the scheduler's method of

working is followed. Despite the fact that the two groups go to work with the same program and receive the same assignments to work out with the program, the expectation is that the two groups will differ in the extent to which they trust the program. Basically, the two groups only differ in the technical vs. non-technical dimension.
The two groups received the same assignments. The assignments, which had to be executed by the subjects, were the same for every subject. The following list shows the assignments each subject received.

1. Open the file: TEST.ZKR
2. Import a person from the file: OUD.TRS
3. Adapt Goossens' personal data card by placing her in Team A
4. Transport 'Molenaar' to a file (choose the name of this file yourself)
5. Show weeks 3 to 7 on the screen
6. Schuurman is ill in week 4, show this in the schedule
7. Give 'Aartsen' Monday, Tuesday and Wednesday off so (s)he can be scheduled in week 4
8. Switch fringe condition 14 (Switch on fringe condition 14)
9. Switch goal function 5
10. Adjust goal function 5 to 1.00
11. Check the fringe conditions
12. Solve the problem found by the system
13. Again check the fringe conditions
14. Generate a schedule for week 5 with the aid of the interactive session
15. Again check the fringe conditions
16. Close ZKR
17. Save the schedule under the name of TEST1.ZKR

On the basis of the six divisions of subtasks, the assignments can be categorised. In section 5.1.2.2, the six subtasks are named and implemented. The subtasks were: administrating, counting, harmonising, valuing, adapting and negotiating. The assignments the subjects had to execute can be categorised in the previous six general sub-scheduling tasks. The following tripartite division can be made regarding the assignments:

Question:	Administrating	Harmonizing	Adapting/ Valuing
1	x		
2	x		
3	x		
4	x		
5	x		
6		x	
7		x	
8	x		
9	x		

10	x		
11			x
12		x	
13			x
14			x
15			x

The assignments themselves consisted of sub-assignments. For example, assignment 14 was an assignment in which the subject had to generate both a day shift and a night shift. The most important parts of the scheduling task and the scheduling system were covered by the assignments. The subjects received a partially completed schedule that was empty at certain places. Therefore the subjects could generate a schedule for the week that was given in the assignment, and fill in some shifts. In the schedule file, a fictitious ward of about 30 employees was specified. The subjects were expected to execute a scheduling task in which the assignments resembled reality as much as possible in terms of order and content.

5.1.4 Location of the tests

Every test took place at the workplace of the subjects. This meant the tests were executed on the wards where the subjects work. The machine on which the program runs is a 486-processor laptop with a colour screen, which the experiment leader took with him. Thus, everyone worked on the same machine so no one was at an advantage.

5.1.5 Subjects of Test I

The subjects who participated in this experiment are all people who have mastered scheduling well. This meant that these persons have all had at least three years of experience in scheduling staff for a nursing ward and that these persons still execute the scheduling task at this moment. Furthermore, the education of the subjects was the same for everyone. The subjects were selected at random from hospitals all over the Netherlands; no hospitals that already work with ZKR were selected. This meant that none of the subjects knew the ZKR system. Hospitals in the cities of Gouda, Arnhem, Oss, Breda, Rotterdam, Leiden, Den Bosch and Nijmegen (in random order) participated in this research. The hospitals all were average to large hospitals, and two hospitals were university hospitals.
The subjects in the non-technical group came from ordinary nursing wards, while the technical group consisted of persons from technical wards like cardiology and Intensive Care.

The subjects in the non-technical group came from the following wards:

Neurology (Nursing ward)	4
First Aid	1
Internal	9
Geriatrics	1

The subjects in the technical group came from the following wards:

IC	8

Cardiology	7

5.1.6 Design and procedure of Test I

The subjects were asked whether or not they wanted to work with the ZKR program. Subsequently, a standard introduction of, at the most, half-an-hour was given on the working of the program. The introduction was standard and was the same for every subject. In thirty minutes, the important elements of the program were explained and within that time, the program can be understood well. From the instruction which has been given to the present users of ZKR, it appears that the program is very user-friendly and that the program is a realistic reproduction of the 'normal' scheduling process which schedule makers almost always go through. Therefore, it became possible to explain the essential elements to new users in the short time of 30 minutes. As mentioned, this training has been standardised and thus is the same for every subject. Then the subjects were given the assignments that they had to execute. The subjects were asked to generate a schedule for a fictitious ward on the basis of the previously discussed assignments. However, this fictitious ward was based on an existing ward, but all names had been changed and none of the subjects knew the ward that was used as an example. Furthermore, the subjects were asked to think aloud during the use of the program. After the execution of the assignments, the subjects were asked to fill in an electronic questionnaire, which was developed with the aid of the EDWIN basic 2.0 program.

5.1.7 Measurements of Test I

While working with the ZKR program, every action of the users was registered in a logfile in which the following actions were recorded: the identification number of the action, the total time since starting up the program and the time between two actions. Furthermore, the subject was asked to think aloud and every remark the person made was recorded on tape in order to be analysed with the aid of the system of categories. Finally the subjects were subjected to an electronic questionnaire, which was developed with the aid of the EDWIN basic 2.0 program. This questionnaire actually consists of five different questionnaires. The questionnaires cover the following subjects: personal details, computer expertise, computer anxiety, computer attitudes and trust. In the following sections of this chapter, an explanation is given of why these five questionnaires were used.

5.2 Results

5.2.1 Reliability test of the questionnaires

First an examination was made of whether or not the questionnaires were reliable on item level. Therefore, the SPSS for Windows reliability was done. The following table indicates the results of the reliability analysis for three questionnaires (anxiety, attitudes and trust).

Test	N of Cases	N of Items	N of Deleted	Alpha
Trust	30	16	5	.7297
Anxiety	30	26	0	.9268
Attitude	30	13	7	.8281

Table 5-1: reliability test of the questionnaires

The result of the analysis of the trust questionnaire was that some questions (5) were removed from the trust questionnaire. The Alpha of the final trust questionnaire was .7297 for 30 persons and 16 final questions. This Alpha is good enough for 16 questions to be able to state that the reliability of the test is sufficient.

The Alpha for the anxiety test was .9268, which was extremely good for 26 items. Maurer (1983; 1984; 1994) based the anxiety questionnaire on an existing questionnaire, namely, CAIN (Computer Anxiety Index). The results of this research confirm the value of this anxiety questionnaire. The Alpha for the attitude questionnaire was .8281, where 7 questions were removed from the original questionnaire. This Alpha also can be called high and the questionnaire was thus sufficiently reliable to be used. The attitudes questionnaire was also based on an existing questionnaire, namely, CAS (Computer Attitude Scale) by Nickell and Pinto (1986).

5.2.1.1 Conclusion

The reliability of the questionnaires was good. The Alpha for the three scales could be called good and no additional questions had to be removed from the questionnaires. In the following section, an examination is made of the normality of the four scales of trust in order to clarify whether or not z-scores can be generated and which statistical tests can be used.

5.2.2 Normality of the four trust scales

Because the different measures, which are registered, have quite different scales, it can be interesting to transform the scores into so-called z-scores, after which the z-scores can be combined into one measure. In the transformation to z-scores, it becomes possible to create one measure that indicates a score on trust. A condition for transforming to z-scores is that it has to be assumed that the scores are divided normally. To find out whether or not this was true, a test was executed in order to examine whether or not the scores on the questionnaire were divided normally. Moreover normality is of great importance for various statistical techniques.

The results of these tests are represented below:

	Group1 (NT)		Group2 (T)	
	skewness	Kurtosis	skewness	kurtosis
TRQUEST	-.377	.525	-,368	,332
TRCATEGO	-.177	-.450	-,110	-1,615
TRBEHAV	.638	.523	-,926	-1,065
TRREACT	-.622	-.080	-1,048	1,738

Table 5-2: normality of the four trust scales

5.2.2.1 Conclusion

All the above measures were not divided normally for the two groups. This meant the measures could not be transformed into z-scores and thus that it was not possible to generate one measure by means of z-scores. Furthermore, the results indicated that, because of the abnormal division of the measures, only non-parametric tests could be used.

5.2.3 Correlation between group and the variables: age, gender, computer expertise, computer anxiety and computer attitudes

The four trust scales are not supposed to measure anything else than the aspect of trust. Therefore other important aspects which may possibly have a disturbing influence on the measured trust have to be examined. It is necessary to find out whether there are variables that possibly interfere with the results of the four scales. This topic will be discussed in the following sections. Here, it is important that the four different variables which can possibly interfere are divided reasonably over the two groups, and that the four scales of trust cannot significantly correlate with the variables that are possibly interfering. If this were the case, a variable that is not divided well over the two groups could have a disturbing influence on the results of the measured trust. Thus, First an examination is made of the correlations between group membership and the variables of age, gender, computer expertise, computer anxiety and computer attitudes.

```
Correlations
       GENDER       AGE    COMPEXP      ANXIETY      ATTITUDE
Spearman's rho      Correlation Coefficient
GROUP  ,202         -,066  ,264         -,294        -,035

Sig. (2-tailed)
GROUP  ,285         ,730   ,166         ,115         ,855
```

Table 5-3: correlation between group and: gender, age, computer expertise, computer anxiety and computer attitudes

The 'Group' variable is not significantly related to the variables of age, gender, computer expertise, anxiety and attitudes. This means that the variables of age, sex, computer expertise, anxiety and attitudes do not have a relevant influence on the four trust scales.

5.2.4 Correlation between group membership and scales of trust

5.2.4.1 Introduction

This section deals with the way in which the four scales of trust and the GROUP variable are connected. This is done by calculating the correlation between the scales of trust and the group membership. Someone in group 1 does not work on a technical ward, while someone in group 2 does. Therefore, it can be expected that the four scales of trust have a positive significant relation with group membership. This means there should be a linear correlation between GROUP and each of the four scales of trust.
In the following table the results of the performed analysis are given.

	GROUP	TRBEHAV	TRCATEGO	TRQUEST	TRREACT
Spearman's rho	Correlation Coefficient				
GROUP	1,000	,576	,695	,494	,142
TRBEHAV	,576	1,000	,274	,359	,285
TRCATEGO	,695	,274	1,000	,543	-,070
TRQUEST	,494	,359	,543	1,000	-,067
TRREACT	,142	,285	-,070	-,067	1,000
Sig. (1-tailed)					
GROUP	,	,000	,000	,002	,226
TRBEHAV	,000	,	,083	,026	,063
TRCATEGO	,000	,083	,	,001	,364
TRQUEST	,002	,026	,001	,	,362
TRREACT	,226	,063	,364	,362	,

Table 5-4: correlation between group membership and scales of trust

The above table indicates that GROUP and TRQUEST show a good correlation, where the relation ($p \leq .002$). The correlation between GROUP and TRCATEGO is .695 which can be called a strong correlation, where this relation is strongly significant ($p \leq .000$). The correlation between GROUP and TRREACT is almost nil, namely .142 (p =≤ .226). This means there is not a significant correlation between TRREACT and GROUP. This will be dealt with in more detail later. As the final important matrix cell, the correlation between TRBEHAV and GROUP can examined. The correlation between these two variables can be called strong and significant, namely .576 ($p \leq .000$). The only variable that does not have a significant relation with the GROUP variable is the TRREACT variable. This means that membership of a specific group has a strong relation with the measure of trust measured on the TRQUEST, TRCATEGO and TRBEHAV scales.

In order to know whether the four scales of trust are not influenced to an important extent by other factors, some important factors that can exercise a disturbing influence will be investigated with respect to the correlation between these factors and the four scales of trust. These factors in this research are: age, gender, computer expertise, education, computer anxiety and computer attitudes.
The different correlations in Table 5-3 show that, between these variables and the GROUP variable, there are no correlations worth mentioning or there are correlations that are only slightly significant. The background education was approximately the same for everyone. From this, it can be derived that the groups do not differ much on the variables, and thus that the scores on the variables are divided reasonably

homogeneously over the two groups (technical and non-technical). The previous correlations between group membership and the variables indicate that the five variables are well divided over the two groups. The two groups do not differ on the score of the six variables and thus can be regarded as groups which are equal on the six variables.
Now the relation between the four scales of trust and the above five variables can be examined.

5.2.4.2 Age

First, the correlation between AGE and the four scales is examined. It is expected that AGE will not show a significant correlation with the four scales of trust. The following table gives the correlation between AGE and the four scales of trust.

Correlations	TRBEHAV	TRCATEGO	TRQUEST	TRREACT
Spearman's rho		Correlation Coefficient		
Age	,213	-,238	-,169	,152
Sig. (2-tailed)				
Age	,259	,231	,373	,423

Table 5-5: correlation between AGE and the four scales

It can be said the AGE variable does not have a strong influence on each of the four scales. It can be stated that AGE is not a strong moderator variable on the total score of the measures of trust.

5.2.4.3 Gender

Gender has always been an important subject in research. In this case the relation between GENDER and the four scales of trust is examined. The results are indicated in the following table.

Correlations	TRBEHAV	TRCATEGO	TRQUEST	TRREACT
Spearman's rho		Correlation Coefficient		
GENDER	,354	,416	,200	,089
Sig. (2-tailed)				
GENDER	,055	,031	,290	,639

Table 5-6: correlation between GENDER and the four scales

The above table indicates that TRCATEGO correlates significantly. This means the variable TRCATEGO possibly experience a disturbing influence from the GENDER variable. The correlation does not approach the correlation found with GROUP; nevertheless, it can be expected that GENDER can be a disturbing variable. The variable TRBEHAV is also significant and could possibly also have a disturbing influence. GENDER does not correlate with membership of a specific group while TRCATEGO and TRBEHAV do correlate highly with GROUP. Probably, GENDER is divided reasonably equally over the two groups, therefore GENDER does not have a disturbing effect on the data.

5.2.4.4 Computer expertise

Because the research was done with a computer, it is possible that COMPUTER EXPERTISE could be a variable that influences the results. In the following table, an overview is given of the correlations between COMPEXP and the four scales of trust.

Correlations	TRBEHAV	TRCATEGO	TRQUEST	TRREACT
Spearman's rho Correlation Coefficient				
COMPEXP	,128	,349	,152	,373
Sig. (2-tailed)				
COMPEXP	,509	,080	,430	,046

Table 5-7: correlation between computer expertise and the four scales

COMPUTER EXPERTISE, then, is not related to the four scales of trust. This means the COMPEXP variable does not have a significant influence on the four scales of trust. It is remarkable that TRREACT does correlate significantly (p = .023). This is not much, however, and it is possible that the small correlation between TRREACT and GROUP can be explained by the COMPEXP moderator variable. TRCATEGO also correlates significantly (p = .040) with COMPEXP. This correlation is also not very strong, but might be a moderator variable. The other two scales are not significantly correlated to COMPEXP.

5.2.4.5 Computer anxiety

The ANXIETY scale should show a small negative correlation with the four scales, because trust is a mechanism to decrease anxiety. ANXIETY is strongly correlated with a lack of control and because trust is a mechanism to increase control, a high trust will give a low ANXIETY score. In the table below, the correlation of anxiety with the four scales of trust is shown.

Correlations	TRBEHAV	TRCATEGO	TRQUEST	TRREACT
Spearman's rho Correlation Coefficient				
ANXIETY	,037	-,396	-,297	-,082
Sig. (1-tailed)				
ANXIETY	,423	,020	,055	,333

Table 5-8: correlation between computer anxiety and the four scales

Only the TRBEHAV scale does not show a correlation with the ANXIETY scale. The scale TRCATEGO does correlate significantly but is relatively small (-.396). The remaining scales of trust indicate a slight negative correlation; the expectation was indeed that this would be negative. In fact, all scales are correlated negatively, as was the expectation. This means the ANXIETY variable does not have a significant influence on the four scales of trust although, as was mentioned in Chapter 3, there is a slight negative relation with ANXIETY.

5.2.4.6 Computer attitudes

Also the ATTITUDE scale should show a slight correlation with the four scales of trust. In this case, the correlation should be slightly positive. In principle, there should be a correlation because it can be assumed that people who have a negative attitude towards computers will not trust computers, and that people who have a positive attitude towards computers are more likely to trust computers.

```
Correlations
             TRBEHAV      TRCATEGO     TRQUEST1     TRREACT
Spearman's rho      Correlation Coefficient
ATTITUDE     -,172        ,184         ,025         -,094

Sig. (1-tailed)
ATTITUDE     ,183         ,180         ,498         ,311
```

Table 5-9: correlation between computer attitudes and the four scales

The ATTITUDE scale does not show a large correlation with the four scales of trust. This means the ATTITUDE variable does not have a significant influence on the four scales of trust.

5.2.4.7 Conclusion

The correlations between the four scales and the five variables, measured in relation to the expectations, is represented in the following table. A '+' stands for "according to expectation" and the '-' stands for "not according to expectation".

	TRBEHAV	TRCATEGO	TRQUEST	TRREACT
AGE	+	-	+	+
GENDER	-	-	+	+
COMPEXP	+	+	+	-
ANXIETY	+	+	+	+
ATTITUDE	+	+	+	+

Table 5-10: expectations of correlations between the four scales and the five variables

It is appears to be plausible that the four measures of trust indeed do measure trust. The measures of trust measure trust, and thus something other than the five variables of age, gender, computer expertise, computer anxiety and computer attitudes. The TRREACT scale of trust is, in this case, slightly problematic because it seems to measure computer expertise more than trust.
Thus, it has been checked that the scales measure what they now measure. The hypotheses from Chapter 3 can now be tested.

5.2.5 Difference between technical and non-technical

5.2.5.1 Introduction

In order to get to know whether the two groups also differ on the four scales of trust, a test has to be executed which gives a definite answer regarding the hypothesis to be tested. The starting point of the hypothesis is that there is a difference between the two groups (technical and non-technical) in the amount of trust they have in a new, complex computer program.
The four scales measure trust. Therefore, in this case, the H0 hypothesis is:

> *There is **no** significant difference in the score between the technical and non-technical group on the four scales of trust.*

The H1 hypothesis then reads:

> *There **is** a significant difference in the scores between the technical and non-technical group on the four scales of trust, in which the technical group scores significantly higher on the four scales of trust.*

From the previous H0 and H1 hypotheses, it appears that, on the basis of groups, which are defined a priori, either there is or there is not a difference in score between these two groups.

Because earlier on in this chapter it became clear that the measures are not divided normally, it is not possible to execute a discriminant analysis or MANOVA. The technique of analysis that now has to be used is the previously discussed Mann-Whitney U-test.

5.2.5.2 Differences in trust between the two groups (technical and non-technical)

In this study, four dependent variables that are not divided normally and two groups (technical and non-technical) are investigated. Table 5-4 shows the correlation of the different scales of trust with the group variable.
There is the question of the chance capitalisation when the test is carried out on an alpha of 5%; therefore, the alpha has to be divided by four. Thus, the new alpha for the four scales of trust becomes 1.25%. This means that the alpha-values have to be under the 0.0125 in order to be significant. Table 5-4 shows that the first three scales (TRBEHAV, TRCATEGO and TRQUEST) are very significant and that the last scale of the four scales of trust (TRREACT) is not significant. That the TRREACT scale is not significant may be caused by the previously mentioned moderator variable of computer expertise (COMPEXP).

5.2.6 Difference in types of trust

5.2.6.1 Introduction

This section deals with the differences between the types of trusts as they are identified in Chapter 3 paragraph 4.1. That is to say not the difference in trust, faith and confidence, but the differences in what is trust based upon. The expectation is that less familiarity with computers/programs will lead to a lack of familiar applicable strategies (Trust II). Furthermore, the expectation is that less familiarity will lead to a lack of logical operations on familiar parts of an actor (Trust III). Therefore, a person is forced to fall back on the strategy of obtaining trust by consulting external actors (Trust I). A person who has little familiarity with the actor cannot fall back on previously gained strategies that have won trust. These strategies are the basis of the type II trust. Furthermore, it will be difficult for persons with a lack of familiarity to execute logical operations on the available data (Trust III), because the data are simply not available as a result of the unfamiliarity. The route, which still can be followed, is to fall back on the possibility of asking another actor questions (for example, a person or a system). This is the type I trust. Conversely, it then should be the case that persons who have built up a reasonable familiarity can make more use of trust of the types II and III. It can be expected that the technical group will score higher on trust of the types II and III. The following section discusses type I trust.

5.2.6.2 Trust I

Type I trust can be distinguished in the subjects' protocols. These are actions that are oriented towards asking for help from the system or the test supervisor. This category is scored on item number 13 in the list of categories in Chapter 4. Item number 13 comprised remarks that involved asking direct help from the test supervisor. Advice was often requested when a person did not know to do in order to execute an assignment. It can be expected that a person who has reasonable trust in a program will not ask for much help from the test supervisor initially, but will try to find out him/herself how something works. This lack of trust, illustrated by asking advice from the test supervisor on more than one occasion, is expressed in the score on item 13 of scores list. The following expectation can be formulated:

> *It is the expectation that the non-technical subjects will score significantly higher on the type I trust than the technical subjects.*

The following table shows the result of the Mann-Whitney test.

Item	Mean Non-Tech.	Mean Tech.	Exact 1-Tailed P
13	19.50	8.89	.0002

Table 5-11: Trust I

The above table indicates that there is a significant difference between the score of the non-technical group and that of the technical group. The average shows that the non-technical group asked for advice more often than the technical group. This appears to support the expectation that less familiarity leads to a lack of familiar applicable strategies (Trust II) and of logical operations on familiar parts (Trust III), forcing a person to fall back on an attempt to stabilise Trust I by asking advice from an external actor. As mentioned, Trust I is the strategy of getting trust by means of consulting external actors.

5.2.6.3 Trust II and III

Type II trust can be distinguished in the subjects' protocols. These are actions that are executed without thinking. This category can be scored on item number 15 in the list of categories in Chapter 4. Item number 15 refers to actions which were executed directly, without help. It can be expected that a person who has a reasonable trust in a program will not ask the test leader many questions initially, but will first try to find out him/herself how something works. In the protocols, no distinction could be made between the trust types II and III. Therefore, it can be assumed that, when an assignment was executed, this was done on the basis of trust type II. Trust type II is expressed in the scoring on item 15 of the scores list. The following expectation can be formulated:

> *It is the expectation that the technical subjects will score significantly higher on trust type II than the non-technical subjects.*

The following table shows the result of the Mann-Whitney test.

Item	Mean Non-Tech.	Mean Tech.	Exact 1-Tailed P
15	7.81	19.75	.0000

Table 5-12: Trust II and Trust III

The above table indicates that there is a significant difference between the scores of the non-technical and the technical group. The averages illustrate that the non-technical group makes less use of trust type II than the technical group. This means the expectation that more familiarity leads to an increase of familiar applicable strategies (Trust II) can be maintained.

5.2.6.4 Conclusion

The difference between people in the degree of familiarity with an actor (technical equipment) results in people making use of different types of trust.
The more familiar a person is with an actor, the more this person will make use of trust type II.
The less familiar a person is, the more this person will make use of trust type I, namely, consulting external actors.

5.2.7 Development of trust

5.2.7.1 Introduction

In Chapter 3, an outline is given of how the development of trust could take place over time. An important element was that trust increases and decreases in the course of time and this is a result of the events that occur during the interaction with an actor who is to be trusted. This is indicated in Figure 3-3 by the change in the degree of trust at certain points in time. An event which influences trust negatively is, for instance, the occurrence of an event where an actor produces a negative result not expected by the trusting person. An event can also have a positive effect on trust, when the result of an actor's action produces a clearly positive result for the trusting person. Furthermore, the figure indicates that as long as there is no counter-evidence and the actor keeps on working as it should, trust will increase until the trusting person starts seeing the evidence for trust as comprising 100% certainty.

The following sections describe the reaction times that were recorded with the aid of the logfiles. This measure of trust, in the form of speed, does not seem a very reliable measure for trust when studying trust at a group level, because it is influenced too much by computer expertise. Someone with much computer expertise has, regardless the size of trust, a quicker interaction with the system than someone who has less computer expertise. Thus, the aspect of computer expertise mainly causes the difference in speed, and therefore the effects of trust are difficult to measure. Because speed is applied here per subject, the variable of computer expertise for both variables, which are to be examined, are the same and only the influence of trust remains. This means that when the subjects' data are compared, the influence of computer expertise is too large to be able to make statements on the measure of trust. However, if the speed which one and the same person scores on different parts is compared, the influence of computer expertise is then filtered out. The influence of computer expertise on one part is exactly the same as on the other part. The fluctuations in speed then can be ascribed to the fluctuations in trust. In the following sections, this will be elaborated further.

5.2.7.2 Influencing trust negatively

This section discusses how trust is influenced negatively. Trust is influenced negatively when the actor to be trusted produces an unexpected result. An unexpected result is defined as a result that appears when the user of the program had expected another result. When, for instance, a trusting person gives the actor to be trusted a command, this person has a certain expectation of what the actor will do. It may be possible that this command is not executed by the actor in the way the person intended. At this moment, the actor does something the person had not expected. This violation of the expectations of the trusting person will have consequences for the trust in the program. Therefore, the reaction time of the user will increase. In a figure, this is indicated as follows:

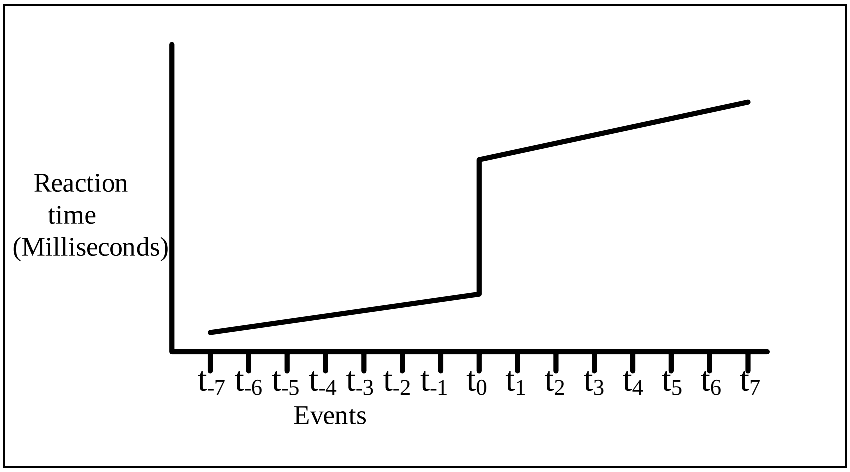

Figure 5-5: negative action

At time t_0, an event that reduces trust takes place. Therefore, a difference appears between the average reaction time before time t_0 and the average reaction time after time t_0. During the tests with ZKR, these negative remarks almost all referred to the same negative event. In the assignments, the subjects were asked to check the fringe conditions. Almost every negative event came into being because the subjects clicked on the menu option 'Switch fringe conditions' while they meant 'Check fringe conditions'.

In order to be able to show that trust is reduced after the appearance of this type of unexpected event, this event has to be identified with the aid of the protocols. When such an event appears in a protocol, the logfile indicates where this event takes place. The time at which the event takes place is thus determined, and examination can be made of whether or not the speed before the event differs from the speed after the event. The seven reaction times are examined before and after the event. Seven reaction times are chosen because, on the whole, it can be stated that the seven reaction times before and the seven reaction times after an event can be seen as a unity in order to be able to observe a brief effect. Furthermore, it appeared that with the seven reaction times no overlap between the two events (negative or positive) was found. These data were registered from each of the protocols and were subsequently analysed with the aid of the paired T-test. The expectation was that the technical group would not have fluctuations that can be measured in the amount of trust, because this group already has a certain trust in technical products, and therefore is not very willing to relinquish that trust lightly. Those in the non-technical group, however, have not yet invested much trust in technical products and are therefore more willing to adapt the trust to negative events.
The results are shown in the following tables. First Group 1's table (non-technical group) is displayed.

Group 1: Non-technical

Paired Differences					
Mean	SD	SE of Mean	t-value	df	1-tail Sig
-2487,30	15270,320	4828,899	-,52	9	,309
95% CI (-13411,0; 8436,429)					

Table 5-13: Reaction time (before and after negative event for group 1)

The above table indicates that the average reaction time before and the average reaction time after the event do not differ significantly. Therefore, it cannot be demonstrated that the trust of the non-technical group is significantly reduced.
The following table displays the paired T-Test data for Group 2 (Technical); here, the averages of the reaction times before the negative event are compared to the average reaction time after the negative event.

Group 2: Technical

Paired Differences					
Mean	SD	SE of Mean	t-value	df	1-tail Sig
5498,750	8441,330	4220,665	1,30	3	,142
95% CI (-7933,29; 18930,79)					

Table 5-14: Reaction time (before and after negative event for group 2)

The above table indicates that the average reaction time before and the average reaction time after the event do not differ significantly.
Next to investigating the effects within groups of a negative event on trust also the differences between groups on the effects of a negative event should be investigated. The following table shows the result of the correlations between group membership and the difference between reaction times before and after the event:

```
Correlations
                DIFF
Spearman's rho      Correlation Coefficient
GROUP          ,196

Sig. (1-tailed)
GROUP          ,251
```

Table 5-15: correlations between group membership and difference between reaction times before and after negative event

There is no significant relation between group membership and effects of the negative event.
Therefore it cannot be demonstrated that the trust of the technical group is reduced significantly. Perhaps the program is just too good and fits in too well with the users' mental models to register any effect.
In relation to these data, it is remarkable that, in the non-technical group, the number of persons who experience negative events is much higher than for the technical group (10 compared to 4); therefore it seems the program quite meets the expectations of the technical group, or at least these members do not judge the events negatively, while for the non-technical group more events are experienced as being negative.

Here, the problem may be that the program worked well, while influencing trust negatively only appears with, for instance, a clear mistake in the program. In the protocols, the most negative event, which could influence trust, was the appearance of an unexpected result of a user's action. It is possible that the events, which are scored here, do not have enough influence on the subjects' trust. It is possible that the trust will decrease measurably when a mistake appears which has serious consequences. In general, people are good at dealing with unexpected events and have more difficulty when dealing with clear errors or shortcomings.
In the following section, the appearance of a positive event is discussed.

5.2.7.3 Influencing trust positively

This section deals with how trust is influenced positively. Trust is influenced positively when the actor to be trusted gives a positive expected result and the user expresses his or her appreciation of this. Imagine a program executes an action at time *t* which is just completely what the user expects, and he/she notes this consciously. After this event, the trust will be increased and therefore the reaction time will be decreased. In a figure this can be represented as follows:

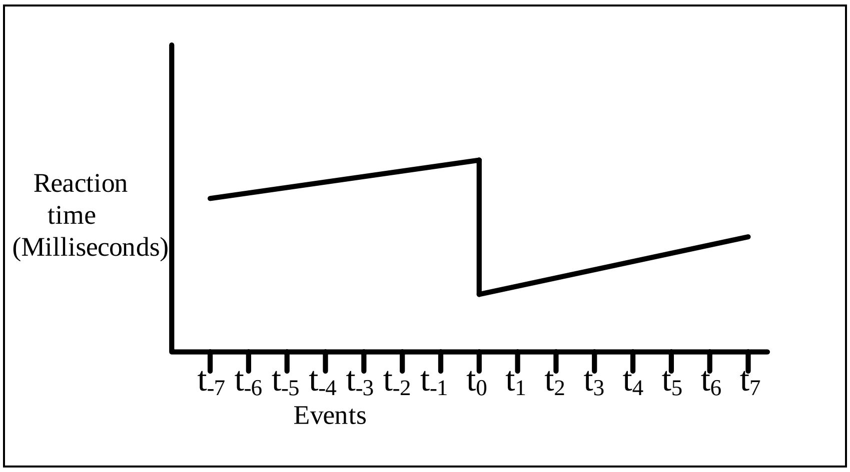

Figure 5-6: Positive action

At time t_0, an event that increases trust takes place. A difference in the average reaction times before and after the positive event occurs, where the reaction time after the event decreases.

In order to be able to show that the trust increases after the appearance of a positive event, this event first has to be identified with the aid of the protocols. When, in a protocol, an event of this type appears, the logfile shows where this event took place. As was explained in the previous section, the time at which the event took place is determined, and thus one can investigate whether or not the reaction time before the event differs from the reaction time after the event. In our study, the seven reaction times before and after the event are examined. From each of the protocols these data are registered and subsequently analyzed with the aid of the paired T-Test. The expectation is that the reaction times of the non-technical group will become significantly better, while the reaction times of the technical group will not improve significantly. The technical group already has a certain amount of trust in the technically high-quality product.

The results are shown in the following tables. Group 1's table (non-technical group) is shown first.

Group 1: Non-technical

Paired Differences					
Mean	SD	SE of Mean	t-value	df	1-tail Sig
-3398,00	16021,738	5664,540	-,60	7	,283
95% CI (-16792,5; 9996,508)					

Table 5-16: Reaction time (before and after positive event for group 1)

The above table indicates that the average reaction times before and after the event do not differ significantly. Therefore, it cannot be shown that in Group 1 (non-technical) the trust becomes significantly larger.

The following table shows the paired T-Test data for Group 2 (technical), in which the averages of the reaction times before and after the positive event are compared.

Group 2: Technical

Paired Differences					
Mean	SD	SE of Mean	t-value	df	1-tail Sig
-6808,75	13385,744	6692,872	-1,02	3	,192
95% CI (-28108,5; 14490,96)					

Table 5-17: Reaction time (before and after positive event for group 2)

The above table indicates that the average reaction times before and after the event do not differ significantly.

Next to investigating the effects within groups of positive events on trust also the differences between groups on the effects of positive events should be investigated. The following table shows the result of the correlations between group membership and the difference between reaction times before and after positive events:

```
Correlations
             DIFF
Spearman's rho      Correlation Coefficient
GROUP        -,102

Sig. (1-tailed)
GROUP        ,376
```

Table 5-18: correlations between group membership and difference between reaction times before and after positive event

There is no significant relation between group membership and effects of positive events. Therefore, it cannot be shown that the trust of the technical group is significantly reduced.

In relation to these data, it is remarkable that, in the non-technical group, the number of persons who experience positive events is much higher than for the technical group (8 compared to 4). Therefore, it appears that the program reasonably meets the technical group's expectations of the program, or at least they do not judge the events positively, while, for the non-technical group, more events are experienced as being positive or remarkably good. In the protocols, the most positive event that could influence trust was the program executing an assignment well, where the user expressed his/her appreciation of this aloud.

5.2.7.4 Trust over a period of time

When someone makes use of a program and this program shows behaviour that is positively appreciated by the user, the trust in the program will increase. This means the trust increases during use if not too many trust-undermining events take place. In order to be able to check this, one can look at whether or not the reaction time in the initial stage of using the program is slower than that at the end of using the program. The subjects executed a standard assignment, in which almost every sub-assignment for the subjects was unique. This means the effect of learning can be excluded almost completely, because every sub-assignment was new to the subjects. One can hardly speak of a repetition of the various assignments. The differences in reaction time data and the observation data between the initial and the final stages of performing the

assignments therefore cannot be ascribed to an effect of learning. However, they can be ascribed to an increase of trust.
If, then, a difference in reaction times and observation behaviour data is found, this will be based on an increase of trust. For both groups, there will be a difference in increase. The technical group will experience little or no measurable increase because it can be assumed that a certain amount of trust has already been achieved, while the non-technical group has to build up the trust completely during working with the program.
The following table shows the result of the comparison of the reaction times at the beginning and at the end of the program. A paired T-Test is again used. The results are shown in the following tables. Group 1's (non-technical) table is examined first.

Group 1: Non-technical

Paired Differences					
Mean	SD	SE of Mean	t-value	df	1-tail Sig
12506,08	9713,087	2507,908	4,99	14	,000
95% CI (7127,151; 17885,01)					

Table 5-19: Reaction times (beginning and end for group 1)

The above table indicates that there is indeed a significant difference in reaction times in the initial and at the final stages of using the program. If a person has more interactions with the actor to be trusted, the trust will increase. The difference in the reaction times at the beginning and end of the assignment is significant. For the non-technical group, a clear increase of trust is visible.
As stated, during working with the program, the trust will not increase as much for the technical group as for the non-technical group. In order to be able to examine this, a paired T-Test is again executed. The following table shows the data for the technical group.

Group 2: Technical

Paired Differences					
Mean	SD	SE of Mean	t-value	df	1-tail Sig
962,4200	13905,824	3590,468	,27	14	,396
95% CI (-6738,37; 8663,209)					

Table 5-20: Reaction times (beginning and end for group 2)

The technical group does not show a significant difference in reaction times.
Again the differences between the two groups should be investigated. The following table shows the results of a Spearman's rho correlation between the group variable and the difference between the beginning and end reaction times:

```
Correlations
               DIFF
Spearman's rho      Correlation Coefficient
GROUP          -,474
Sig. (1-tailed)
GROUP          ,004
```

Table 5-21: correlation between the group variable and the difference between the beginning and end reaction times

There is a strongly significant relation between group membership and the difference between the reaction times of the beginning and the end of using the program.
This means that the assumption that the technical group will have a higher level of trust, and thus fewer fluctuations in trust, can be confirmed. If an increase in speed were to be a result of an effect of learning, both groups should have shown an increase or no increase in speed. The fact that only one group shows this increase indicates that the increase in speed is not the result of a possible effect of learning.
The following section discusses the differences in trust when executing certain scheduling tasks with the knowledge-based system ZKR 1.2.

5.2.8 Difference in tasks and measure of trust

5.2.8.1 Introduction

In Chapter 3, in which the model of trust was discussed, it appeared that trust mainly becomes of importance when the trusting person runs a certain risk. When there is no risk, that is, when there is no trust to be betrayed, there is mention of confidence. The trusting person knows for certain that the actor will behave in a fixed way. Furthermore, it is easier for a person to place trust when the negative result of betrayal is not so large. One can say that the higher the risk experienced by the trusting person, the more difficult it is to win trust. With the use of a program, the part that does not entail a very large risk will be trusted most easily, while a part which entails a large risk will not be trusted so readily. The beginning of this chapter dealt with the assignments and the link with the theories on the area of scheduling and the subtasks within that area. The assignments could be divided into subtasks as follows:

1. Assignments 1 to 5 and 8 to 10 are administrative assignments.
2. Assignments 6,7 and 12 are adaptations, in fact, of an almost empty schedule.
3. The assignments 11 and 13 to 15 are harmonising assignments, combined with valuing.

Because the risk with the various parts was not equal, the amount of trust will differ on the different parts. Furthermore, a difference in trust will appear between the two groups. The non-technical group will have a problem with estimating the risk and will tend to estimate everything on the same level of risk. The administrative actions will be more familiar to the technical group than to the non-technical group in relation to the adaptations and harmonisations. This means that the difference in trust expressed in reaction times between the administrative or adapting assignments and the harmonising assignments should be larger for the technical group than for the non-technical group. The non-technical group sees fewer differences between the different tasks. The non-technical group has not built up familiarity with entering and

adapting data in a technically high-quality product. This group has never needed to adjust and adapt parts using technical devices, after which one had to trust that the data in the device were correct. The technical group, however, has often adjusted devices and has had to trust that the devices have been well adjusted. What the technical group had not previously experienced was that a device took over a certain cognitive complex action. This means that, in the technical group, the trust in the administrative part was proportionally larger than the trust in the harmonising. Furthermore, this means that, in the non-technical group, the trust in the administrative part does not differ proportionally from the trust in the harmonising part.
In the following part, the results of the T-Tests on the trust scores on reaction times are shown. The scores of the non-technical group (Group 1) are shown first.

Group 1: Non-technical

```
         Paired Differences       |
    Mean          SD    SE of Mean |      t-value            df      1-tail Sig
 ----------------------------------|--------------------------------------------
-3859,79    8353,295      2156,812 |        -1,79            14           ,047
 95% CI (-8485,69; 766,107)        |
```

Table 5-22: Paired T-Test on reaction times (administrative and adapting)

```
         Paired Differences       |
    Mean          SD    SE of Mean |      t-value            df      1-tail Sig
 ----------------------------------|--------------------------------------------
1846,330    9476,518      2446,826 |          ,75            14           ,231
 95% CI (-3401,59; 7094,250)       |
```

Table 5-23: Paired T-Test on reaction times (administrative and harmonizing)

```
         Paired Differences       |
    Mean          SD    SE of Mean |      t-value            df      1-tail Sig
 ----------------------------------|--------------------------------------------
-2013,46    6094,301      1573,542 |        -1,28            14           ,110
 95% CI (-5388,38; 1361,447)       |
```

Table 5-24: Paired T-Test on reaction times (adapting and harmonizing)

From Table 5-22 to Table 5-24, it appears that, in terms of the reaction times, no difference is found between trust in the various parts of the program. In the technical group, a difference in the execution of the various tasks should be found, in which the largest difference should be found in the average reaction time between administrative tasks and harmonising and the average reaction time between adapting and harmonising. The results of the reaction times for the various tasks are shown in the following tables.

Group 2: Technical

```
Paired Differences       |
    Mean          SD    SE of Mean |      t-value            df      1-tail Sig
 ----------------------------------|--------------------------------------------
-852,913    4628,446      1195,060 |         -,71            14           ,243
 95% CI (-3416,06; 1710,235)       |
```

Table 5-25: Paired T-Test on reaction times (administrative and adapting)

Paired Differences					
Mean	SD	SE of Mean	t-value	df	1-tail Sig
-9682,13	7812,980	2017,303	-4,80	14	,000
95% CI (-14008,8; -5355,44)					

Table 5-26: Paired T-Test on reaction times (administrative and harmonising)

Paired Differences					
Mean	SD	SE of Mean	t-value	df	1-tail Sig
-8829,22	8154,255	2105,420	-4,19	14	,000
95% CI (-13344,9; -4313,54)					

Table 5-27: Paired T-Test on reaction times (adapting and harmonizing)

Next to the effects of tasks within the groups (technical and non-technical) the effects between groups should be investigated. The following table shows the results of the between group correlations. The variables ADMADAP, ADMHARM and ADAPHARM stand for respectively the difference between administrating and adaptation, administrating and harmonizing and the difference between adaptation and harmonizing.

Correlations	ADMADAP	ADMHARM	ADAPHARM
Spearman's rho		Correlation Coefficient	
GROUP	-,289	-,504	-,458
Sig. (1-tailed)			
GROUP	,061	,002	,005

Table 5-28: effects between groups (administrating and adaptation (ADMADAP), administrating and harmonizing (ADMHARM) and the difference between adaptation and harmonizing (ADAPHARM))

The only non significant relation is the relation between group and ADMADAP. ADMHARM and ADAPHARM show significant results.

The tables indicate a clear difference between the average reaction times in the tasks of administrating and harmonising and between the average reaction times in the tasks of adapting and harmonising. Thus, it appears that, for the non-technical group, trust does not have a specific character in which every differing part has a similar amount of trust. This does not apply to the technical group. The technical group shows a difference in the amount of trust in the different parts. The parts with which the subjects in the technical group were reasonably familiar show an essential difference with the parts with which the subjects were not familiar. The part with which the subjects were not familiar was the harmonising part, where the so-called actor characteristic of application is the main element.

5.3 Conclusion

In this chapter, after explaining the test procedure used, the results were discussed. First, a reliability test was done for the three questionnaires (TRUST, ANXIETY and ATTITUDES). It appeared the questionnaires had a high reliability score and the remaining questions were used in processing the test material. Furthermore, examination was carried out on the influence of group membership on the variables: Age, Gender, Computer expertise, Education, Computer anxiety and Computer

attitudes. It appeared that membership of a particular group did not have a significant influence on the variables.
It appeared that this group membership was strongly correlated to the TRCATEGO, TRQUEST, and TRBEHAV scales of trust. The TRREACT variable did not have a significant relation with group membership. In addition, examination was carried out on whether or not the scales of trust were also influenced by the variables: Age, Gender, Computer expertise, Computer anxiety and Computer attitudes. The following table expresses the expectations of the correlations between the four scales and the five variables.

	TRBEHAV	TRCATEGO	TRQUEST	TRREACT
AGE	+	-	+	+
GENDER	-	-	+	+
COMPEXP	+	+	+	-
ANXIETY	+	+	+	+
ATTITUDE	+	+	+	+

Table 5-29: the expectations of the correlations between the four scales and the five variables

A '+' stands for “according to expectation” and the '-' stands for “not according to expectation". This chapter also discussed the difference between the technical group and the non-technical group in terms of this statistic. It indeed appeared that for the three scales: TRCATEGO, TRQUEST and TRBEHAV, the hypothesis could be affirmed, while, for TRREACT, the hypothesis could not be affirmed.
Examination was also carried out on the difference between the two groups in terms of the type of trust used. It appeared that the technical group made more use of the TRUST II type of trust and that the non-technical group made more use of the TRUST I type of trust. Furthermore, it appeared that none of the groups showed a reduction in trust after the appearance of a negative event, nor did either of the two groups show an increase in trust after a positive event. Additionally, it appeared that, as expected, the non-technical group showed the largest increase in trust over the course of time. Finally, the difference in trust between the two groups during the various types of subtasks was examined. It indeed appeared that there is a difference between the two groups in the degree of trust shown during the subtasks. The non-technical group appeared to be quite constant in the trust, while the technical group clearly showed less trust when program parts, which showed clear actor characteristics, had to be dealt with.
The question posed in Chapter 1, concerning whether or not measures can be identified which can measure trust, can be answered positively. Trust can be measured by four measures, where it must be stated that, in mutually comparing the measures of trust, the reaction times can be influenced to a too great extent by the measure of computer expertise. The measure of reaction time can be used to measure fluctuations in trust within the subject. The fluctuations in trust are investigated within a session with one subject. It can be stated that the first hypothesis from Chapter 3 is confirmed by the results of the research. The first hypothesis was:

> *A significant difference in the scores between the non-technical group and the technical group on the four trust scales is present, where the technical group has proportionally more trust in the program than the non-technical group.*

It indeed appears to be the case that the technical group has more trust in the program than the non-technical group.

In Chapter 7, these findings are elaborated in more detail.

6. Test II (CASH)

6.1 Introduction

This chapter describes the tests with the commercial and the non-commercial test groups. First, an explanation is given of how the tests were carried and which measurements were executed. Subsequently, the results of the tests are presented. This chapter describes the testing of Hypothesis 2 and links the data found to the theoretical frame outlined in Chapter 3. In the following sections, the method of Test II is dealt with.

6.2 Method of Test II

6.2.1 Introduction to Test II

This chapter describes the testing of the second hypothesis, as outlined in Chapter 3. This essence of this second hypothesis was concerned with whether or not the subjects' goals were fulfilled by the system. This meant that two groups had to be selected, namely, a group for which the system did not fulfil every goal and a group for which every goal was fulfilled. In Test II, the second hypothesis is tested, namely:

> *When a knowledge-based system does not support every important goal a user has, trust in the knowledge-based system will be lower than when every goal that a user has is supported.*

The previous hypothesis can be converted into the following testable hypothesis:

> *A significant difference in scores between the non-commercial group and the commercial group is present on the four trust scales, where the commercial group has proportionally more trust in the program than the non-commercial group.*

As was stated in Chapters 3 and 4, two groups of users are required in order to be able to test this hypothesis. For the first group, not every goal is supported while, for the second group, every goal is supported. In this case, the CASH knowledge-based system is used, where the first group consists of commercial users and the second group consists of non-commercial users. When using CASH, the non-commercial group is chiefly interested in having the program select a good hearing aid, while the commercial group also has to consider the factor of the price when selecting a hearing aid. This does not mean that the most expensive aids always have to be chosen. It is also possible that, when two good aids are selected, the user may advise buying the cheaper one as a form of service to the customer.

6.2.2 The stimulus material of Test I (the CASH program)

In this test, the subjects used a program that selects, interactively with the user, hearing aids for people with a hearing disorder. The system used is the CASH diagnosis system (Numan, 1991; 1994; 1994c). CASH is a system that selects hearing aids for people with a hearing disorder on the basis of the loss of hearing and other personal details. The CASH knowledge-based system was developed in a non-commercial environment (Groningen Academic Hospital) and does not take the user's commercial interests into consideration when selecting a hearing aid. Before the system can be described, the background of the CASH system and the theoretical background of the program will first be dealt with.

6.2.2.1 Background to CASH

In the Audiology department of the Groningen Academic Hospital, there arose the need for a knowledge-based system for supporting the fitting of hearing aids to people with a hearing disorder. In the Audiology department, fitting a hearing aid to people with a loss of hearing is done by different experts who select an aid on the basis of two types of audiograms and the patient’s personal details. During the fitting process, the expert has to select aids from a total set of more than a hundred devices. Every expert who handles this fitting applies his/her own rules, which have not been recorded anywhere, and of which the expert may not even have explicit knowledge. Furthermore, as far is known, there is little or no similarity between the various experts in terms of the hearing aids that are actually selected. The first version of CASH (Numan, 1991) was made using one of the first versions of Knowledge Pro under Windows. This version had few graphical possibilities for interaction with the user. Therefore, the program was reprogrammed in Microsoft C (Numan, 1994b). With this, the interaction with the user was improved to a large extent.

6.2.2.2 Theoretical background to CASH

In this subsection, the theoretical background to CASH is dealt with. In order to give an impression of which part of the process of fitting a hearing aid is supported by the program, a description of the route most patients follow in acquiring a hearing aid, and a brief description of fitting an aid, are given below.

6.2.2.2.1 The patient’s route

First, of course, a complaint arises. The patient or the patient’s relatives begin to complain about his/her bad hearing. The patient then goes to a general practitioner who performs a simple hearing test. When the general practitioner thinks the patients hearing is defective, a referral to an Ear, Nose and Throat (ENT) specialist or to an audiological centre follows. When the patient has a complaint that is not too complex, the specialist will perform a hearing test, prescribe an aid and refer the patient to an expert working in a shop or store where they sell hearing aids (this session often lasts about 15 minutes).

When the complaint is too complicated for the specialist, he/she will refer the patient to an audiological institute. In the audiological institute, more elaborate tests are done, a hearing aid is tried and then he/she is referred to the expert working in a store where they sell hearing aids.

The expert in a store where they sell hearing aids makes a made-to-measure earpiece and supplies the prescribed aid. Six weeks after fitting, the patient returns to the person who has fitted the aid in order check whether or not the aid is satisfactory. If not, the procedure is started again.

6.2.2.2.2 Fitting

Generally, the fitting process is as follows:
The person who first fits the aid to the patient (mostly an ENT specialist) obtains the patient's case history containing various data about the patient that can be of importance for the person who fits the hearing aid to the patient. After this, the patient is called up by his/her doctor, and the person who fits the hearing aids subjects the patient to one or two audiograms. Often the patient is subjected to two audiograms, namely: the tone threshold, and audiograms of speech. The tone threshold is based on being able to hear tones. The speech-distinction audiogram is based on repeating a pronounced word correctly. Generally, there is some small talk between the patient and the person who fits the hearing aids, in which the latter can obtain some details from the patient him/herself. Now the person who fits the hearing aid to the patient selects some hearing aids, which, in conjunction with the patient, are adjusted for amplification level, pc, agc, etc.

6.2.2.2.3 Theoretical background to hearing aid selection

In general, much of the scientific research on selecting hearing aids for people with a loss of hearing (Wit, 1985; Groen, 1971; Green et al., 1989a; 1989b) has been done by taking as the point of departure the fact that the selection of a hearing aid is based on the two types of audiograms (tone threshold and speech discrimination).
In the process of knowledge acquisition, it became clear that the data from the audiograms alone are not sufficient for fitting a good hearing aid to the patient. It appeared that the experts use different data, which are scarcely mentioned in the literature. These data were: age, the former brand of aid, the living conditions, the patient's preferences for a certain type of aid, the patient's manual skills, ENT problems and varying loss of hearing. During the implementation of the system, the data, which are required over and above the audiogram information, are taken into consideration.
Selecting hearing aids has the character of making a diagnosis. In the chapter on the tests with ZKR, an overview was given of the types of subtasks that have to be executed in order to come to a solution to a question. Because different subtasks require a different measure of trust, a survey of a division of the diagnosing task is given.
First, the subtask of *identifying* can be distinguished. Here, one attempts to see what exactly is the problem. This is the problem identification.
Then the subtask of *making an inventory* is executed. Here, an attempt is made to establish which data have to be distinguished in order to be able to make a statement on a diagnosis, and on which variables influence the problem. Making an inventory of the task may lead to a document in which the data are stored, just like a case history for a patient.
After this, the *selection* subtask is executed. Selection means that, from the total set of diagnoses, those diagnoses are selected which possibly apply to the problem.
Finally, the *valuing* subtask is executed. The *valuing* subtask is, in fact, the selection of the best candidate diagnosis from the set of possible diagnoses. At this point, the commercial point of view mainly appears. The commercial group will bear in mind

the commercial interests, while the non-commercial group will not take these into consideration.
In the following section, the CASH program itself is discussed.

6.2.2.3 The CASH program

6.2.2.3.1 The knowledge-based system part

It will be clear the system has to use different data in order to be able to make a good selection of aids. From the knowledge acquisition it appeared, as expected, that the experts not only made use of the audiograms to which the patient was subjected but also of different types of personal details. The system then had to bear these data in mind as well. The most important of these are listed below. The system takes the following aspects into consideration when selecting hearing aids:

- The tone threshold audiogram (for amplification and frequency information).
- The audiogram of speech (for amplification, gain control and peak clipping).
- Personal details (namely: age, the former brand of hearing aid, the living conditions, the patient's preferences for a certain kind of aid, the manual skills, ENT problems and varying loss of hearing).

On the basis of this information, a hearing aid is selected and offered to the user.

6.2.2.3.2 The user interface

The parts of the system mentioned have to be presented to the users in some form. The medium with which the users interact with the computer is the user interface. This is discussed in the following section.

6.2.2.3.2.1 Making an inventory

First, the audiograms have to be entered. In order to develop this system, different theoretical points of view were made use of. One of these involved using the 'world metaphor' of Mulder et al. (1992) (see also Chapter 2). Here, one begins from the knowledge the user has of the world in order to have the user relate to the user interface. The user can apply the knowledge he/she has of the world to the aspects of the interface. Therefore, the learning period is shortened and the user has the feeling that he/she 'knows' the interface. For a more elaborate analysis, see Numan (1994c).
In the following two figures, both input screens are shown. Here, a value is inserted by means of a simple click on the mouse button in graphs, which are identical to the old common graphs on paper; the system uses these to be able to handle the values scored by the patient. In the interface, an attempt is made to transfer the representation on paper into a screen representation. Therefore, the user can fall back on knowledge that was already present before the system was used. The following figure displays the speech-discrimination audiogram.

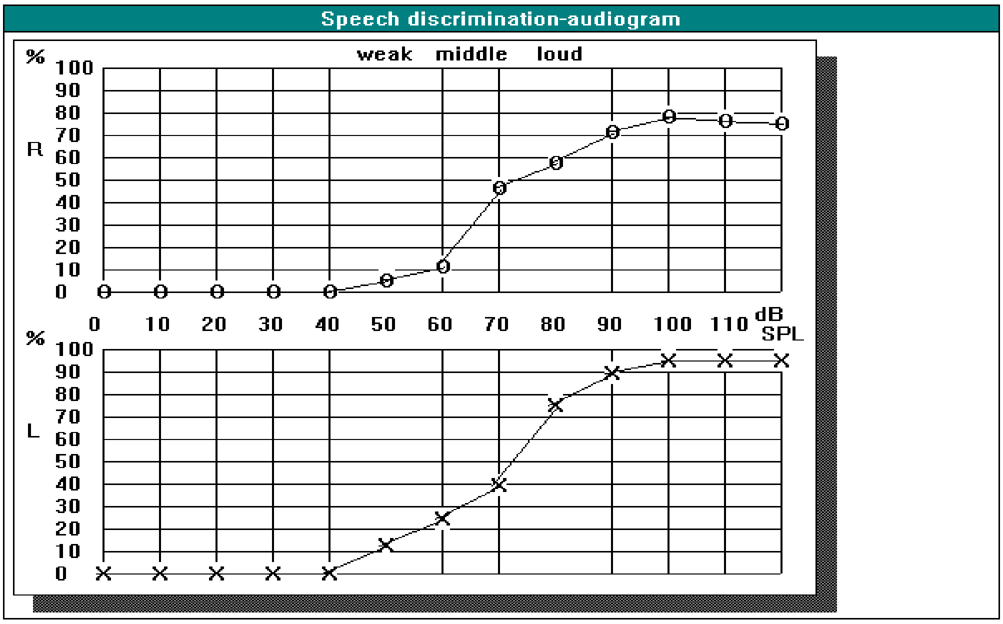

Figure 6-1: speech-discrimination audiogram entry window

Filling in occurs as follows: when a patient with the right ear at a sound intensity of 80 dB can still repeat 60% of the words uttered, the mouse is clicked at the intersection of 80 dB (horizontal) and 60% (vertical). At that moment, an 'O' appears at the place at which the mouse has been clicked. Then a line is drawn from the score on 70 dB and 90 dB. In this way, the speech-discrimination audiograms are filled in.

Here a clear use of the world metaphor can be seen. During the design of the system, use was made of the old task concepts to support the new task concepts. The input has the character of manipulating an object (the curve) with reversible actions (Shneiderman, 1992). Because the emphasis is on this, the future user needs to understand less of new syntactic and semantic computer knowledge, and therefore the program has a closer relation with the user's environment. The following figure presents the threshold audiogram.

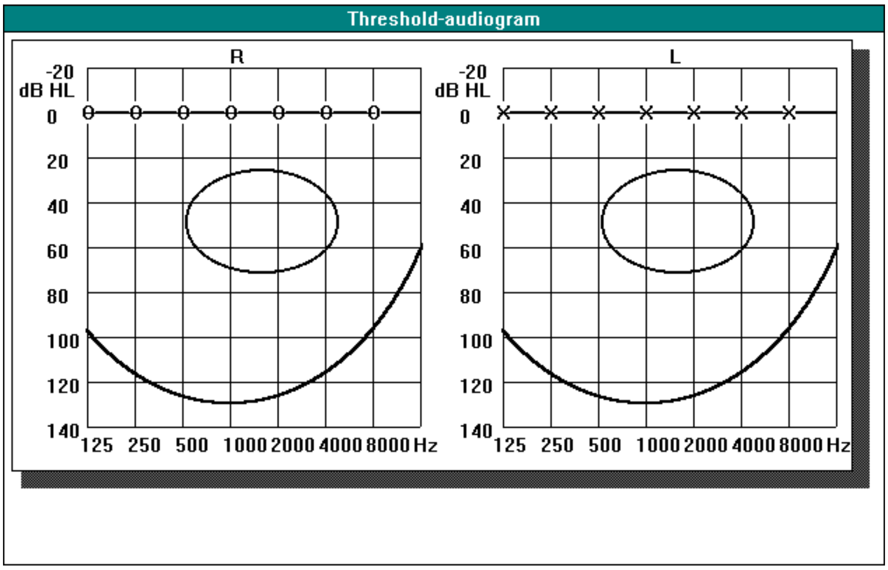

Figure 6-2: threshold audiogram entry window

Filling in takes place as follows: when a patient, with the right ear, at a pitch of 250 Hz and at a sound intensity of 40 dB, still hears the tone issued, the mouse is clicked on the intersection of 250 Hz (horizontal) and 40 dB (vertical) in the left graph. At that moment, an 'O' appears at the point where the mouse has been clicked. Then a line is drawn from the score on 125 Hz and the score on 500 Hz. In this way, the threshold audiogram is completed.
Next to the audiograms, which have to be entered, there is also the possibility to insert personal details. The program can make use of these details so that the selection of hearing aids becomes more detailed than in cases when only audiograms are used. The questions are also completed with a simple click on the button. The following figure shows a part of the questionnaire, as it has to be filled in by the user.

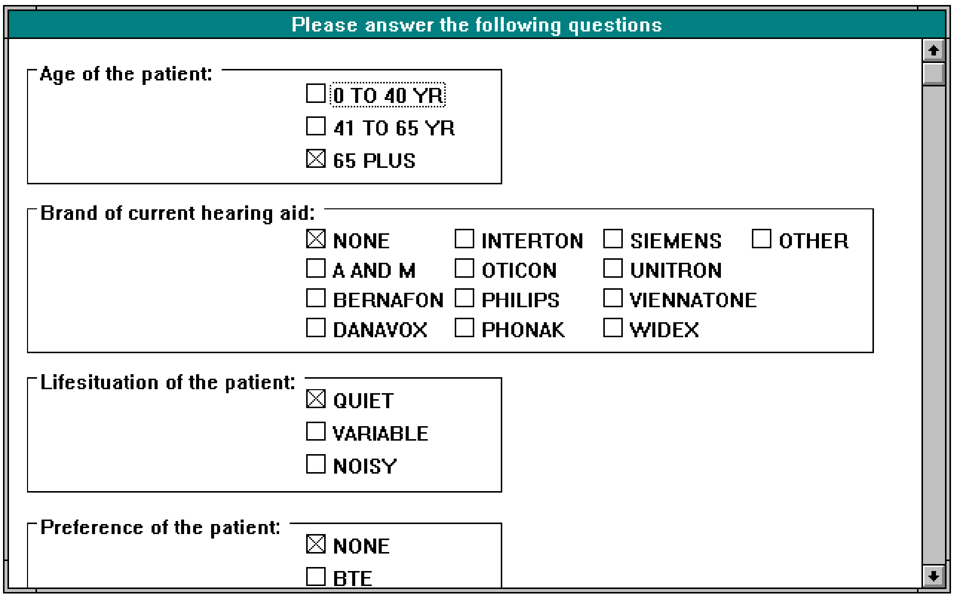

Figure 6-3: questionnaire window

When these details have been filled in, the hearing aid can be selected.

6.2.2.3.2.2 Selection and appreciation

When all the required details have been entered, the CASH knowledge-based system gives a list of solutions in terms of the aids which, combined with the personal details, are good enough for the specified hearing. The following figure displays a list of solutions as offered by CASH.

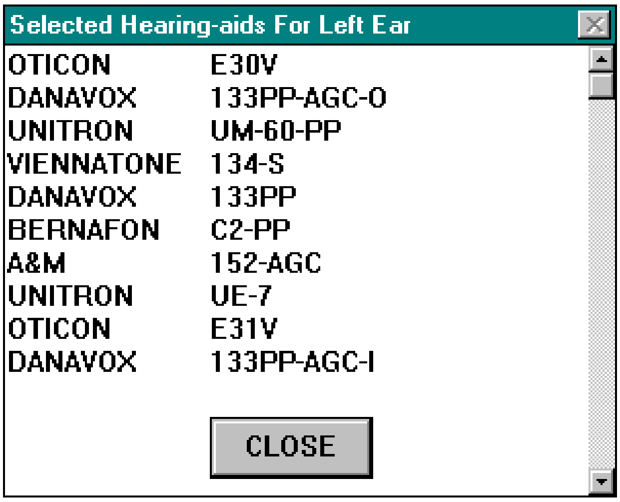

Figure 6-4: selected hearing-aids

The above figure displays how the list with selected aids is presented to the user. The aids are arranged from top to bottom, the most suitable aid is at the top of the list, and the least suitable aid is at the bottom. However, every aid can be adjusted. The user can click on an aid in the list if he/she wants to, and CASH will then show the suitability of the aid in a new window. The following figure displays this window.

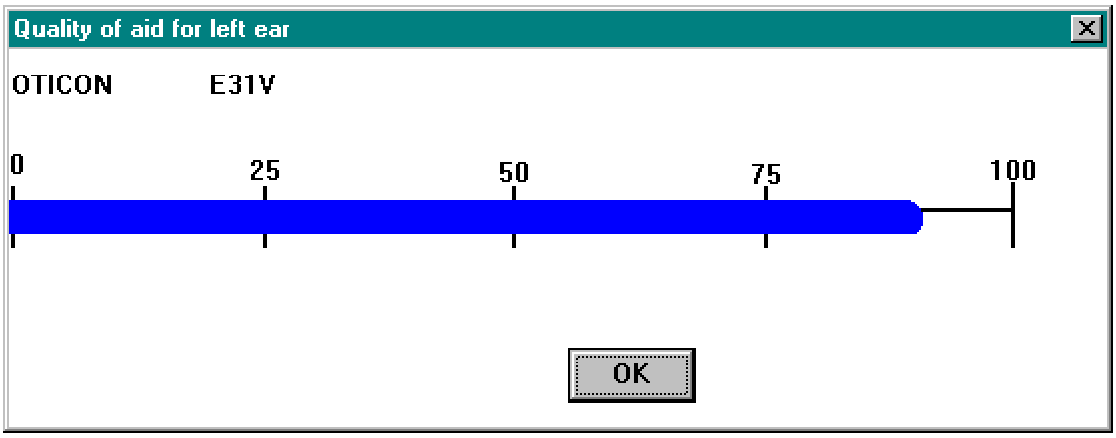

Figure 6-5: quality of aid window

The above figure displays the quality of the aid for the specified hearing. In the above figure, the quality score for the OTICON E31 is almost 90%. This means that the aid meets the demands made by the current hearing on a hearing aid for almost 90%.

6.2.2.4 Conclusion

The CASH program is a program that can be learnt quickly by a new user. The interface is extremely user-oriented and during the development of the user interface, much attention was paid to the user's method of working.

6.2.3 Assignments for the subjects

Every subject received the same assignment. The subjects had to enter a case history in the system twice and select aids with the system. Then the subjects had to select one or more aids from the list. The following list shows which sub-assignments the user had to execute per patient who had to be entered.

1. Fill in the threshold audiogram for the left ear.
2. Fill in the threshold audiogram for the right ear.
3. Fill in the speech-discrimination audiogram for the left ear.
4. Fill in the speech-discrimination audiogram for the right ear.
5. Fill in the personal details.
6. Use the system to select hearing aids.
7. Choose a hearing aid.

The assignments 1 to 5 are *inventorying* assignments. Assignment 6 is a *selection* assignment and 7 is a *valuing* assignment. The selection was executed by CASH and not by the user. This means that, in principle, two types of assignments are executed, namely, inventorying and valuing assignments.

6.2.4 Subjects in Test II

The subjects in the first group are users who have not worked with CASH before and who have a commercial background. In this case, every expert working in a store in Groningen where they sell hearing aids was approached and every one of them (n=8) co-operated.

The subjects in the second group are users who have not worked with CASH before and have a non-commercial background. In this case, each of the 8 hearing therapists of the University Hospital Groningen (AZG) was approached and every one of them (n=8) co-operated. Both groups are quite similar to each other in terms of background (education, age and expertise). On the point of task expertise, every subject can be called an expert.

6.2.5 Locations of Test II

Every test took place at the subjects' place of work. This meant the subjects in the commercial group were subjected to the tests in their shops. The person who administered the tests took along a computer on which the subjects could work. The subjects in the non-commercial group were subjected to the tests in the ENT department of the Groningen Academic Hospital. The tests were administered using the computer that was also used in the test with the ZKR system.

6.2.6 Design and procedure of Test II

The subjects were asked whether or not they wanted to work with the CASH program. Then a standard introduction of ten to fifteen minutes was given about the working of the program. The introduction is a standard introduction and was the same for every subject. In ten to fifteen minutes, the important elements of the program were explained; within that time the program can be understood quite well. Then the subjects were given two case histories (which were selected from a thousand case histories at random), with information on the patient that had to be entered by the subject. The input consisted of the personal details of the case histories (patients) and of two audiograms for each ear per patient. In entering the input, each case history was finished completely before a new case history was entered. Furthermore, the subject was asked to think aloud during working with the program. After the assignments were executed, the subjects were asked to fill in an electronic questionnaire which was developed with the aid of EDWIN basic 2.0.

6.2.7 Measurements of Test II

During working with the CASH program, all input entered by the user was recorded in a logfile in which the action performed, the total time since the start up of the program and the time in-between two actions was recorded. From these files, information can be obtained on reaction times and executed actions. Furthermore, the subjects were asked to think aloud and each of the subjects' remarks was recorded on tape, in order to be transcribed later and analyzed with the aid of the system of categories. After the test, every subject was subjected to a questionnaire with the aid of EDWIN basic 2.0.

6.3 Results

6.3.1 Reliability test of the questionnaires

First an examination was made of whether the test was reliable at item level. Therefore, the reliability test was done with SPSS for Windows, resulting in some of the questions from the trust questionnaire being removed. The following table shows the three final reliability test results of the three questionnaires.

Test	N of Cases	N of Items	N of Deleted	Alpha
Trust	16	13	8	.7074
Anxiety	16	26	0	.8969
Attitude	16	16	4	.7913

Table 6-1: reliability test of the questionnaires

The alpha of the final trust questionnaire was .7074 for 16 persons and the final 13 questions. This alpha is good enough to be used as a measure for 13 questions. The alpha for the anxiety test was .8969, which was extraordinarily good for 26 items. In this test, the same anxiety test (Maurer, 1983; 1984; 1994) was used as in the test with ZKR. The alpha for the attitude questionnaire was .7913, with which 4 questions were removed from the original questionnaire. This alpha, too, can be called high, meaning that the questionnaire is sufficiently reliable to be used. In this test, Nickell and Pinto (1986) used the same attitude test as in the test with ZKR, namely, CAS (Computer Attitude Scale).

6.3.1.1 Conclusion

The above alphas for the three questionnaires are amply sufficient to be able to say that the questionnaires are sufficiently reliable to be used at item level.

6.3.2 Normality of the scales

Because the various measures, which have been registered, have quite different scales, it would have been interesting to make so called z-scores of the scores. As has previously stated, a precondition for transformation into z-scores is that the scores are divided normally. To find out whether or not this was the case, a test was executed to examine whether or not the scores on the trust questionnaire were divided normally. Chapter 5 has already outlined the importance of data being divided normally in order to execute statistical techniques.
The result of these tests are represented here below:

	Group1 (C)		Group2 (N-C)	
	Skewness	kurtosis	skewness	kurtosis
TRQUEST	-.150	-1.568	.764	.875
TRCATEGO	-.142	-1.596	-.337	-.687
TRBEHAV	-.074	-2.613	-2.149	5.145
TRREACT	-.109	-1.463	.749	.412

Table 6-2: normality of the scales

When checking the normality, it is of importance that the scores on the kurtosis and the skewness are near to nil. As can be seen in the table, the skewness on the different scales of Group 1 can be called quite good; however, the kurtosis of the scales is not very acquiescent. The scale measures for Group 1 are not divided normally. For Group 1, the normality of the TRQUEST, TRCATEGO and TRREACT scales is not too bad but, unfortunately, cannot be called normal. The TRBEHAV scale is certainly not divided normally.

6.3.2.1 Conclusion

Each of the measures above is not divided normally for the two groups. This means the measures cannot be transformed into z-scores and thus that it is not possible to generate one measure by means of z-scores. The previous results also indicate that, because of the abnormal division of measures, one can only use non-parametric tests here.

6.3.3 Correlation between group membership and the variables: age, gender, computer expertise, computer anxiety and computer attitudes

The four scales should not measure anything other than the aspect of trust. Therefore, just as in the previous chapter, it has to be investigated whether or not there are other important aspects that may possibly have a disturbing influence on the trust measured. The correlations between group membership and the variables of age, gender, computer expertise, computer anxiety, and computer attitudes are examined.

Correlations	AGE	GENDER	COMPEXP	ANXIETY	ATTITUDE
Spearman's rho	Correlation Coefficient				
GROUP	-,082	-,405	-,299	,449	-,281
Sig. (2-tailed)					
GROUP	,764	,120	,261	,093	,310

Table 6-3: correlation between group membership and the variables: age, gender, computer expertise, computer anxiety and computer attitudes

The GROUP variable is not significantly related to the variables of age, gender, computer expertise, anxiety and attitudes. This means that the variables of age, sex, computer expertise, anxiety and attitudes do not have a relevant influence on the four trust scales.

6.3.4 Correlation between group membership and the scales of trust (TRBEHAV, TRCATEGO, TRREACT and TRQUEST)

6.3.4.1 Introduction

The following section discusses the way in which the four scales of trust and the GROUP variable are connected. This is done by calculating the correlation between the scales of trust and group membership. Someone in Group 2 is in the non-commercial group, while someone in Group 1 is a member of the commercial group. It can then be expected that the four scales of trust will have a positive significant relation with group membership. This means that there is a linear correlation between group membership and each of the four scales of trust, which means that group membership influences the measure of trust in a newly introduced system.

The following table presents the results of the analysis.

Correlations	GROUP	TRBEHAV	TRCATEGO	TRQUEST	TRREACT
Spearman's rho	Correlation Coefficient				
GROUP	1.000	.095	.380	-.528	.488
TRBEHAV	.095	1.000	.315	-.079	.377
TRCATEGO	.380	.315	1.000	-.085	.713
TRQUEST	-.528	-.079	-.085	1.000	.077
TRREACT	.488	.377	.713	.077	1.000
Sig. (1-tailed)					
GROUP	.	.362	.073	.018	.027
TRBEHAV	.362	.	.117	.386	.075
TRCATEGO	.073	.117	.	.376	.001
TRQUEST	.018	.386	.376	.	.388
TRREACT	.027	.075	.001	.388	.

Table 6-4: correlation between group membership and the scales of trust

The above table shows that TRBEHAV, TRCATEGO do not correlate significantly, while TRQUEST (p=.018) and TRREACT (p=.027) do correlate significantly; here the negative correlation between GROUP and TRQUEST is remarkable. This would mean that the non-commercial group scored less on the questionnaire on trust than the commercial group. The expectation was that the non-commercial group would have more trust than the commercial group. In order to find out whether or not there are variables that possibly interfere with the results of the four scales, further investigation was carried out. This is described in the following sections. Here, it is of importance that the different variables that could possibly interfere are divided reasonably over the two groups, and that the four scales of trust cannot significantly correlate with these possibly interfering variables. Examination has to be made of whether or not the different variables might correlate highly with the four scales of trust. If that is the case, it is possible that this variable is a moderator variable that influences the results. AGE is examined first.

6.3.4.2 Age

The following table displays the results of the correlations between the respondents' age and the four scales of trust.

Correlations				
	TRBEHAV	TRCATEGO	TRQUEST	TRREACT
Spearman's rho	Correlation Coefficient			
AGE	-.189	-.280	.022	-.439
Sig. (2-tailed)				
AGE	.484	.294	.936	.089

Table 6-5: correlations between AGE and the four scales of trust

The above table shows that AGE has a high correlation only with TRREACT. The correlation between AGE and TRREACT is negative, which means that the older the respondent, the slower the reaction time. That reaction time has a relationship with age is not very remarkable and is, in fact, a well-known phenomenon. One can state that the AGE variable does not have a strong influence on the three scales of trust: TRQUEST, TRCATEGO and TRBEHAV, while AGE is possibly a moderator variable on the TRREACT trust scale. Another variable that may possibly influence the four scales of trust is the GENDER variable. This is discussed further in the following section.

6.3.4.3 *Gender*

The following table displays how the GENDER variable and the four scales of trust correlate.

Correlations				
	TRBEHAV	TRCATEGO	TRQUEST	TRREACT
Spearman's rho	Correlation Coefficient			
GENDER	-.529	-.161	.090	-.453
Sig. (2-tailed)				
GENDER	.035	.551	.741	.078

Table 6-6: correlations between GENDER and the four scales of trust

The above table indicates that the GENDER variable does how a significant correlation with the TRBEHAV scale. Therefore the GENDER variable can be seen as a moderator variable fro the TRBEHAV scale.

6.3.4.4 Computer expertise

The following variable that is examined is the COMPUTER EXPERTISE variable. It is possible that, as appeared in the test with ZKR, computer expertise is a moderator variable on the four scales of trust. The correlations are presented in the following table.

Correlations	TRBEHAV	TRCATEGO	TRQUEST	TRREACT
Spearman's rho	Correlation Coefficient			
COMPEXP	-.271	.127	.177	.218
Sig. (2-tailed)				
COMPEXP	.310	.639	.511	.417

Table 6-7: correlations between computer expertise and the four scales of trust

The COMPUTER EXPERTISE variable also does not show a significant correlation with the four scales of trust and thus can be excluded as a possible moderator variable. However, it is remarkable that the TRREACT variable does not correlate significantly with COMPEXP. This would have been possible (see also Chapter 5) because reaction times with a computer can be linked to computer expertise. The more expertise, the quicker the reaction times. The following subsection examines whether or not the scales have a significant correlation with a well-known variable in human-computer research, namely that of COMPUTER ANXIETY.

6.3.4.5 Computer anxiety

COMPUTER ANXIETY as a variable does have some relation with trust and thus it can be expected that the four scales of trust have a negative, but not significant, relation with trust. The following table presents the correlations between COMPUTER ANXIETY and the four scales of trust.

Correlations	TRBEHAV	TRCATEGO	TRQUEST	TRREACT
Spearman's rho	Correlation Coefficient			
ANXIETY	.073	-.024	-.391	-.052
Sig. (1-tailed)				
ANXIETY	.398	.466	.075	.427

Table 6-8: correlations between computer anxiety and the four scales of trust

The above table indicates that the three TRQUEST, TRCATEGO and TRREACT trust scales have a negative tendency and that the TRBEHAV scale shows a slight positive tendency. None of the scales correlates significantly with the scale of COMPUTER ANXIETY. This means the ANXIETY variable is also not a moderator variable.

Another well-known scale in the human-computer research is the scale of COMPUTER ATTITUDE. The following subsection discusses the results of the correlations between computer attitudes and the four scales of trust.

6.3.4.6 Computer Attitudes

The results on the scale of ATTITUDE should correspond slightly positively with the four scales of trust, because a positive attitude might influence trust positively. The scale, however, cannot measure the same as the scales of trust. The following table shows the correlations between the scale of ATTITUDE and the scales of trust.

Correlations	TRBEHAV	TRCATEGO	TRQUEST	TRREACT
Spearman's rho Correlation Coefficient				
ATTITUDE	-.014	.114	.235	.004
Sig. (1-tailed)				
ATTITUDE	.481	.343	.200	.445

Table 6-9: correlations between computer attitudes and the four scales of trust

Only the TRQUEST variable shows a slight positive correlation while the remaining scales of trust do not show a correlation of any significance. The ATTITUDE variable, thus, is not a moderator variable of the influence of group membership on the measure of trust.

6.3.4.7 Conclusion

The four individual scales of trust, do not show an important link with the various possible moderator variables. This means that the four scales of trust measures the degree of trust which a person has when working with a program.

6.3.5 Difference between the commercial and the non-commercial group

6.3.5.1 Introduction

In order to find out whether or not the two groups also differ on the new scale of trust, a test had to be executed to provide a decisive answer regarding the hypothesis in question. The starting point of the hypothesis is that there is a difference between the two groups (commercial and non-commercial) in the degree of trust shown in a new complex computer program. Trust is measured by the scale of trust that is the result of the factor analysis of the four scales of trust. Thus, the **H0 hypothesis** in this case is:

> There is **no** significant difference in score between the commercial and non-commercial group exist on the new scale of trust.

The **H1 hypothesis** then is:

> There **is** a significant difference in score between the commercial and non-commercial group on the new scale of trust. The score on the scale of trust will be significantly higher for the non-commercial group than for the commercial group.

From the previous H0 and H1 hypotheses, it appears that on the basis of groups which are defined a priori, investigation should be carried out on whether or not there is a difference in score between these two groups on the scale of trust.

Because this chapter has demonstrated that the measure is not divided normally, no discriminant analysis or ANOVA can be executed. The technique of analysis, which now has to be used, is the previously discussed Mann-Whitney U-Test. This test is now used to test the above hypotheses.

6.3.5.2 Differences in trust between the two groups (commercial and non-commercial)

In this research, use is made of dependent variables that are not divided normally and two groups (commercial and non-commercial). Table 6-4 shows the correlations of the scales of trust with the group variable.

The only scales indicating a significant relation with group membership are TRQUEST ($p = .018$) and TRREACT ($p = .027$). Because of the correction for chance capitalisation which means that α= 1.25% there is no significant relation between group membership and the scales.

6.3.6 Difference in types of trust

6.3.6.1 Introduction

In this section, the different types of trust, as identified in Chapter 3 paragraph 4.1, are discussed. That is to say not the difference in trust, faith and confidence, but the differences in what is trust based upon.

The expectation is that the less a person’s aims are met by an actor that is to be trusted, the smaller the use of the types of trust as distinguished in Chapter 3 paragraph 4.1. This means the commercial group should score significantly lower on Trust I, Trust II and Trust III than the non-commercial group.

6.3.6.2 Trust I

In the subjects’ protocols, type I trust can be distinguished, as was stated in the previous chapter. The lack of trust is expressed in the score on item 13 of the scores list. The following expectation then can be formulated:

> *The expectation is that the commercial subjects will score significantly lower on the type I trust than the non-commercial subjects.*

The following table shows the result of the Mann-Whitney test.

Item	Mean Non-Com.	Mean Com.	Exact 1-Tailed P
13	7.75	9.25	.2868

Table 6-10: Trust I

The previous table indicates that there is no significant difference in score between the non-commercial and the commercial group. The averages show that the non-commercial group asked for help fewer times than the commercial group. The expectation the non-commercial group would make more use of Trust I than the commercial group must be rejected. The following sections deal with trust type II in more depth.

6.3.6.3 Trust II and III

In the subjects' protocols, type II trust can also be distinguished, as was stated in the previous chapter. Type II trust is expressed in the scores on item 15 of the scores list. The following expectation can be formulated:

> The expectation is that the non-commercial group will score significantly higher on type II trust than the commercial group.

The following table presents the results of the Mann-Whitney test.

Item	Mean Non-Com.	Mean Com.	Exact 1-Tailed P
15	7.81	9.19	.2868

Table 6-11: Trust II and Trust III

The above table indicates no significant difference in score between the non-commercial and the commercial group. The averages show that the non-commercial group makes less use of trust type II than the commercial group. This leads to a rejection of the expectation that when more goals are answered by an actor this will lead to an increase of Trust II.

6.3.6.4 Conclusion

The difference between people in the extent to which the goals which an actor supports does not result in people starting to make use of another type of trust. No differences in the type of trust between the commercial group (not every goal supported) and the non-commercial group (every goal is supported) can be found.

6.3.7 Development of trust

6.3.7.1 Introduction

In this section, the development of trust over the course of time, as explained in section 5.2.7 of the previous chapter, is discussed.

6.3.7.2 Influencing trust negatively

This section deals with how trust is influenced negatively. The expectation is that the non-commercial group will not have measurable fluctuations in the measure of trust. Because it is expected that, for this group, the program will meet every goal and therefore, as has become clear in the discussion of the model of trust, the group will not significantly adjust the degree of trust. The commercial group, however, will expect of the program that not every goal they have will be supported and therefore they will have less trust in the program. The results are presented in the following tables.

Group 1: commercial

```
Paired Differences        ″
    Mean          SD    SE of Mean ″      t-value            df      1-tail Sig
″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″
-1022,90    2462,385      1101,212 ″         -,93             4           ,203
 95% CI (-4080,35; 2034,559)      ″
```

Table 6-12: Reaction time (before and after negative event for group 1)

The above table indicates that the average reaction times before and after the event do not differ significantly. Therefore, it cannot be proved that the commercial group's trust becomes significantly less.
The following table shows the paired T-Test data for Group 2 (non-commercial), where the averages between the reaction times before and after the negative event are compared.

Group 2: non-commercial

```
Paired Differences       "
    Mean          SD    SE of Mean "      t-value            df      1-tail Sig
 """""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""""

1311,000    7994,742      3575,357 "          ,37             4           ,366
 95% CI (-8615,78; 11237,78)      "
```

Table 6-13: Reaction time (before and after negative event for group 2)

The previous table indicates that the average reaction times before and after the event do not differ significantly. Next to investigating the effects within groups of a negative event on trust also the differences between groups on the effects of a negative event should be investigated. The following table shows the result of the correlations between group membership and the difference between reaction times before and after negative events:

```
Correlations
               DIFF
Spearman's rho      Correlation Coefficient
GROUP          ,383

Sig. (1-tailed)
GROUP          ,137
```

Table 6-14: correlations between group membership and the difference between reaction times before and after negative events

There is no significant relation between group membership and effects of the negative event.

Therefore, it cannot be proved that trust becomes significantly less for the non-commercial group.
With respect to these data, it is remarkable that, for the non-commercial group, the number of persons who experience negative events is exactly similar to those in the commercial group who experience negative events (5 to 5). Therefore, it seems that the program meets or does not meet both groups' expectations to exactly the same extent. Here, it might be the case that the program worked well, while a negative influence of trust only occurs with, for instance, a clear mistake in the program. In the protocols, the most negative experience, which could influence trust, was the occurrence of an unexpected event as the result of a user's action. It is possible that the events that were scored here did not have enough influence on the subjects' trust. It is possible that trust does decrease measurably when an error occurs which has serious consequences. As previously mentioned, people are generally good at dealing with unexpected events and have more difficulty in dealing with clear mistakes or shortcomings.
The following section discusses the occurrence of a positive event.

6.3.7.3 Influencing trust positively

This section deals with how trust is influenced positively. The expectation is that the reaction times for the commercial group will become significantly quicker, while the reaction times for the non-commercial group will not quicken significantly. The non-commercial group already has a certain measure of trust.
The results are shown in the following tables.

Group 1: commercial

```
 Paired Differences       ″
    Mean          SD    SE of Mean ″      t-value            df      1-tail Sig
 ″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″
-5955,74   24904,998     10167,423 ″         -,59             5           ,291
 95% CI (-32091,9; 20180,45)       ″
```

Table 6-15: Reaction times (before and after positive event for group 1)

The previous table indicates that the average reaction times before and after the event do not differ significantly. Therefore, it cannot be proved that Group 1's (commercial) trust increases significantly.
The following table shows the paired T-Test data for Group 2 (non-commercial), where the averages between the reaction times before and after the positive event are compared.

Group 2: non-commercial

```
Paired Differences       ″
    Mean          SD    SE of Mean ″      t-value            df      1-tail Sig
 ″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″
-3518,79   10771,884      5385,942 ″         -,65             3           ,280
 95% CI (-20659,3; 13621,68)       ″
```

Table 6-16: Reaction times (before and after positive event for group 2)

The previous table indicates that the average reaction times before and after the event do not differ significantly.
Next to investigating the effects within groups of a positive event on trust also the differences between groups on the effects of positive events should be investigated. The following table shows the result of the correlations between group membership and the difference between reaction times before and after the event:

```
Correlations
              DIFF
Spearman's rho      Correlation Coefficient
GROUP         -,282

Sig. (1-tailed)
GROUP         ,250
```

Table 6-17: correlations between group membership and the difference between reaction times before and after positive events

There is no significant relation between group membership and effects of the positive event. Therefore, it cannot be proved that trust decreases significantly for the non-commercial group.
With respect to these data, it is remarkable that, for the non-commercial group, the number of persons who experience positive events is almost similar those in the commercial group (6 compared to 4). In contrast to the results of ZKR, no difference between the two groups can be found here. In the protocols, the most positive event,

which could influence trust, was the program executing an assignment well, where the user said this aloud.

6.3.7.4 Trust over the course of time

In the previous chapter, it was stated that trust will change in the course of working with a program. The non-commercial group will experience little or no measurable increase because it can be assumed a certain level of trust is already achieved on beforehand, while the commercial group will still have to build up trust while working with the program.
The following table shows the result of the comparison between the reaction times in the initial stages of working with the program and in the final stages. A paired T-Test was again applied. The results are shown in the following tables.

Group 1: commercial

Paired Differences					
Mean	SD	SE of Mean	t-value	df	1-tail Sig
1101,025	4377,662	1547,737	,71	7	,250
95% CI (-2558,79; 4760,842)					

Table 6-18: Reaction times (beginning and end for group 1)

The above table indicates that there is significant difference between the initial and final stages when performing an assignment with the program. The difference between the reaction times at the beginning and the end of the assignment is not significant. For the commercial group, thus, no increase in trust can be seen.
For the non-commercial group, it is expected that trust will not increase during working with the program. In order to be able to examine this, another paired T-Test is executed. The following table presents the data for the non-commercial group.

Group 2: non-commercial

Paired Differences					
Mean	SD	SE of Mean	t-value	df	1-tail Sig
-656,362	7268,702	2569,874	-,26	7	,403
95% CI (-6733,15; 5420,424)					

Table 6-19: Reaction times (beginning and end for group 2)

The non-commercial group does not show a significant difference in reaction times. Now the differences between the two groups should be investigated. The following table shows the results of a Spearman's rho correlation between the group variable and the difference between the beginning and end reaction times:

```
Correlations
             DIFF
Spearman's rho      Correlation Coefficient
GROUP        ,035
Sig. (1-tailed)
GROUP        ,462
```

Table 6-20: correlation between the group variable and the difference between the beginning and end reaction times

There is no significant relation between group membership and the difference between the reaction times in the beginning and the end of using the program.
This means that the assumption that the non-commercial group will have a higher level of trust and therefore fewer fluctuations in trust than the commercial group cannot be confirmed. In the following section, the difference in trust during the execution of certain subtasks with the knowledge-based system CASH will be discussed.

6.3.8 Difference in tasks and measure of trust

Introduction
In Chapter 3, in which the model of trust was discussed, it appeared that trust mainly becomes of importance when the trusting person runs a risk. The beginning of this chapter discussed the assignments and the link with the subtasks within the assignments. Analogous to section 5.2.9, the assignments can be divided into subtasks as follows:

1. The assignments 1 to 5 are *inventorying* assignments.
2. The assignments 6 and 7 are *valuing* assignments.

Because the risk with the different parts is not equally large, the amount of trust in the different parts will differ. Furthermore, a difference in trust will appear between the two groups. Particularly with valuing, the commercial group will, in comparison to the non-commercial group, estimate the risk as being higher. The difference in the average between the inventorying assignments and the valuing assignments will be larger for the commercial group than for the non-commercial group.
At the point at which the hearing aids are selected by the system, some of the commercial group's goals are not borne in mind. At this point, the trust expressed by the commercial group will become less.
The following table presents the results of the T-Tests on the scores of trust on the reaction times.

Group 1: commercial

```
Paired Differences       ″
   Mean          SD    SE of Mean ″      t-value            df      1-tail Sig
″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″″
-17186,7    9822,850      3472,902 ″        -4,95            7           ,001
 95% CI (-25398,9; -8974,64)       ″
```

Table 6-21: Paired T-Test on reaction times (inventorying and valuing)

The above table indicates that, with the aid of reaction times, a difference is found between forms of trust in the program in the two subtasks. This means that the inventorying subtask receives more trust than the valuing subtask.

With the non-commercial group, a difference in the execution of the two subtasks should be found. The following table presents the results of the reaction times for the two subtasks.

Group 2: non-commercial

Paired Differences					
Mean	SD	SE of Mean	t-value	df	1-tail Sig
-24974,3	29968,944	10595,622	-2,36	7	,025
95% CI (-50029,0; 80,364)					

Table 6-22: Paired T-Test on reaction times (inventorying and valuing)

Next to the effects of tasks within the groups (commercial and non-commercial) the effects between groups should be investigated. The following table shows the results of the between group correlations. The variable ADMIVALU stands for the difference between inventorying and valuing.

```
Correlations
       ADMIVALU
Spearman's rho      Correlation Coefficient
GROUP  -,000

Sig. (1-tailed)
GROUP  ,500
```

Table 6-23: difference between inventorying and valuing (ADMIVALU)

There is no significant relation between group and ADMIVALU.
The tables indicate a difference between the average reaction times in the tasks of inventorying and valuing; however, there is no significant difference between the two groups. This means it can not be assumed that the expectation is correct, that the difference in trust between the inventorying assignments and the valuing assignments is larger for the commercial group than for the non-commercial group. However there is a difference in reaction times between the two sub tasks.

6.4 Conclusion

In this chapter, the results of processing the tests with the CASH system were discussed. First, a reliability test was performed on the three questionnaires (TRUST, ANXIETY and ATTITUDES). It appeared that the questionnaires had a high reliability score and the remaining questions could be used in processing the testing material. It appeared that the four scales of trust were not divided normally. After this, just as in Chapter 5, examination was made of the influence of membership of the specific group and the possible moderator variables. It appeared that the group membership did not have a significant influence on the variables.
It appeared that group membership was strongly correlated to the TRQUEST scale of trust. Here, it is remarkable that TRQUEST has a reverse relation with group membership. It seems that the commercial group gave more positive answers on the Trust questionnaire. In addition, examination was made of whether or not the four scales of trust also were influenced by the variables of Age, Gender, Computer expertise, Computer anxiety and Computer attitudes. The following table shows how the expectations between the four scales and the five variables, in relation to the expectations.

	TRBEHAV	TRCATEGO	TRQUEST	TRREACT
AGE	+	+	+	-
GENDER	+	+	+	+
COMPEXP	+	+	+	+
ANXIETY	+/-	+	+	+
ATTITUDE	+	+/-	+	+/-

Table 6-24: expectations between the four scales and the five variables

A '+' represents "according to expectation" and the '-' represents "not according to expectation, while '+/-' represents a correlation which was exactly between expected and unexpected. Subsequently, the difference between the commercial group and the non-commercial group was discussed further. It appeared that indeed the TRREACT scale supported the hypothesis. The remaining three scales of TRCATEGO, TRQUEST and TRBEHAV did not support the hypothesis, while the TRQUEST scale denied the hypothesis.
The difference in the use of the type of trust between the two groups was also examined. No difference in the type of use between the commercial group and the non-commercial group could be made evident. It also appeared that none of the groups showed a decrease of trust after a negative event, nor did either of the two groups show an increase in trust after a positive event. Also no change in the amount of trust appeared in the course of time. Neither of the two groups showed a difference in trust between the initial stages of working with the program and the final stages.
Finally, the difference between the two groups in trust shown during the different types of subtasks was examined. A significant difference between the average reaction times during the inventorying and valuing tasks for the two groups appeared. This means that it can be assumed that there is a differences in trust between the inventorying and valuing assignments. However it can not be assumed that the differences in trust are larger for the commercial group than for the non-commercial group.
The question as to whether or not measures that measure trust can be identified can be answered positively. Trust can be measured by four measures; here, it must be said the questionnaire information does correlate remarkably well to group membership. The questionnaire information points in the opposite direction to that expected. The measure of reaction time can be used to measure fluctuations in trust in the subjects; examination of the fluctuations in trust is carried out during a session with a subject.

It can be stated the second hypothesis can not be confirmed by the results of the research. The second hypothesis was:

> *When a knowledge-based system does not support every important goal which a user has, trust in the knowledge-based system will be lower than when every goal is supported.*

It appears not to be the case that the non-commercial group has more trust in the program than the commercial group.
In Chapter 7 these findings are elaborated further and in more depth.

7. Conclusion

7.1 Introduction

The aim of this research is to clarify what trust exactly is and the way in which way this concept could be used within the research on Human-Computer interaction research. In Chapter 2, we saw that Human-Computer Interaction is more than an interface alone. It appeared that there was a need to obtain insight into the mechanisms that are important in the development of expectations and trust. Within the Human-Computer Interaction field of research, not much attention is paid to trust. Some articles that deal with Human-Computer Interaction have been found. In order to be able to obtain clarity regarding what trust in Human-Computer Interaction actually consists of, the following research questions were formulated in Chapter 1.

1a: In which way can trust be described and used?

In this study, an answer to this research question is given in Chapter 3. There, a definition of trust is given on the basis of a study of the literature.

1b: What is the function of trust?

In relation to the second research question, an answer is also given in Chapter 3. The statement is made that trust is a mechanism to reduce complexity. The basis of trust is that someone acts as if there are only a fixed number of possibilities in the future. Trust is anticipating the future by assuming the future is certain or reasonably certain. In this way, people try to reduce uncertainty about the future. Trust is an expectation that certain aspects of the behaviour of the actor to be trusted will yield positive results for the trusting person.

1c: Is it possible to formulate a model of trust?

In Chapter 3, a model of trust was given. The third research question (question 1c) is answered by the development of the model in this research. This model indicates that there are three sorts of trusts that are strongly linked.

1d: Which measures can be identified to measure trust?

Trust can be measured by four measures. The first element that can be measured is the basic attitude of trust. The second way of measuring is by means of the so-called protocol analysis of the think-aloud protocols during interaction with the actor. These protocols can be scored on negative trust remarks and positive trust remarks. The third measure of trust is based on the fact that interacting with a complex system means that a user allocates certain tasks to the system. Whether or not a subsequent check of these tasks takes place will give the opportunity to obtain an indication of a user’s trust. Finally, the interaction process can be looked at. When a person trusts an actor this will speed up the interaction process.

2a: How is trust in a program built up?

According to this research, trust in an actor is built up and broken down by positive and negative results of the behaviour of the actor to be trusted.

2b: How can the dynamic aspects be measured?

The measure of reaction time can be used in measuring trust within the subjects when investigation into fluctuations in trust is carried out within a session with one subject.

3: Which aspects have influence on a user’s trust in a system?

Two hypotheses, which were mainly related to the third research question, were formulated. The first hypothesis concerned the actor to be trusted. In Chapter 5, this first hypothesis was tested and it can be said that the first hypothesis, as stated in Chapter 4, can be confirmed by the results of the research. The first hypothesis was:

> *There is a significant difference in score between the technical and non-technical group on the four scales of trust, in which the technical group in proportion has more trust in the program than the non-technical group.*

It indeed appears to be the case that the technical group has more trust in the program than the non-technical group. On the basis of a generated scale and on the basis of three of the four scales separately, it could be stated that the technical group has more trust than the non-technical group.

The second hypothesis was related to the goals that are supported by the program (chapter 6). It can be said that the second hypothesis can not be affirmed by the results of the research. The second hypothesis was:

> *When a knowledge-based system does not support every important goal that a user has, the trust in the knowledge-based system will be lower than when every goal that a user has is supported.*

It appeared that the non-commercial group did not show a significant higher trust than the commercial group. The group of which not every goal was supported had less trust in the program than the group of which every goal was supported. It did not appear to be the case that the non-commercial group had more trust in the program than the commercial group. Thus, the second hypothesis can not be accepted on the basis of the results of the four scales of trust. Unfortunately, it cannot be concluded that Hypothesis 2 is true. The four scales are not significant.

7.2 Generality of the results

In this section, the generality of the results of this research is discussed. The research was mainly directed towards the interaction with complex computer systems, where both complex computer systems were knowledge-based systems. Knowledge-based systems have some new characteristics and new development methods, in comparison to more traditional systems. Some of these points will be addressed in the following sections.

7.2.1 Traditional systems versus Knowledge-based system

Over the last twenty years, there has been a considerable change in the types of information systems that have been developed. The same can be said about the characteristics of the various systems. In the beginning of the computer era, the programs were relatively simple and mostly worked in batch mode. Nowadays, the programs are highly complex, are sometimes unpredictable in their output, and currently work in real time. Instead of being slaves, computers have increasingly become partners in problem-solving tasks. This trend has reached its (provisional) climax in the development of knowledge-based systems (KBS) (see Chignell & Parsaye, 1988; Cleal & Heaton, 1988; Lucas & Gaag, 1988), which are used in several areas of application. McGraw (1992) states that a certain fear of technology, which was further strengthened by the use of the name ‘expert system’, has arisen.

The previous section indicated that the character of the new systems has clearly changed. The new systems have started to show more actor characteristics, as was

stated in the previous chapters; the climax of actor characteristics has arrived, at least for the time being, with the knowledge-based system.
Knowledge-based systems have been developed to support complex, ill-defined and rather difficult problem solving. These problems can vary from diagnostic problems or planning or scheduling problems to monitoring and control problems. Solutions to these problems are created in the interaction between the KBS and the user. Chignell (1993) called this ‘co-operative human-machine reasoning” which he defines as follows:

> *“Systems of one or more humans and one or more machines that work together to solve problems and carry out tasks using the shared reasoning capabilities of human and machine”.*

This means that the system functions in a co-operative way and does not follow the user slavishly. Its co-operative character is one of the main respects in which a KBS differs from the more traditional systems. In solving a problem with a KBS, there is knowledge on the user’s side *and* on the side of the KBS. Qualitatively spoken, the knowledge differs from the information that a user gets when using a traditional information system. Working on a task with a KBS means that there is an allocation of sub-tasks to both parties: the KBS and the user. This co-operative character of a KBS means that the interface becomes increasingly important. The user interface is the communication medium, pre-eminently between the internal (inference) processes of the system and the user’s reasoning processes. Bødker (1989, 1991) calls this: ‘through the interface”. The fact that the interface is the intermediary means that the interface is the communication channel through which the user’s knowledge will be imported into the KBS and that it is the communication channel through which the knowledge of the KBS is presented to the user. This situation has important consequences for the development of the interface. It means that, when designing the interface, the designer has to pay attention to the knowledge that has to be represented in the user interface. From this it follows that the separation of the interface and knowledge in the system - we could also call them presentation and representation - is not as sharp as is in traditional systems with an interface and data or information.
The role of the interface in a KBS is not only to present a solution in a proper way but also to present the way the system obtained its solution. The system not only helps the user in generating a solution, it also helps with the problem-solving process itself. This last aspect is a much more complex and abstract task in itself (McGraw, 1992).
This all means that the development of knowledge-based systems cannot be done in a linear way as is common with standard information systems.

7.2.2 Traditional system development versus Knowledge-based system development

The previous section indicated that, as a result of the changing character of the new systems, the methods of system development should also acquire another character. In general, the standard methods of system-development have a linear character. Roughly stated, a task analysis and a data analysis are initially performed in order to determine the system requirements. Then the system is built via a functional description and a technical specification. After that, a system is presented to the users who then receive training in how to use the system. Because of the co-operative character of the new systems, it will be clear that after the systems have been built in this way they may no longer have a direct connection to that which the user needs. First, the system will start taking over cognitive tasks, and therefore the user’s needs may possibly change. This change of tasks can be taken care of by approaching the development route in a more interactive way. For a more elaborate analysis of the change of tasks during the introduction of knowledge-based system, see also Numan et al. (1994e). Secondly, the new systems have become so complex that the user ascribes actor characteristics to the system. Therefore, during the development and during the introduction route, the actor character of the new systems should be taken into account. This means that certain characteristics of the future system have to be given attention. Characteristics like trust are of essential importance, as appears from this research, during the introduction of new systems. During the development phase, one can look at how trust in a system can be supported and, during the introduction of the new system, one can explicitly try to support the trust in the new system.

7.2.3 Conclusion

The results of the tests described in Chapters 5 and 6 were based on tests with two different systems. Both were knowledge-based systems. One system (ZKR) was a planning system while the other system (CASH) was a diagnostic system. The outcome of this research can be generalised to the whole domain of complex systems (Weir, 1991). A complex system (such as ZKR and CASH) has several characteristics. One of the main characteristics is that the system gives the user a feeling of co-operation (Chignell, 1993). This means that the user has the feeling that the system is co-operating and that it has a degree of freedom of action. As a result of this, it can be expected that the outcomes of this research also apply to complex systems in general. Diagnostic systems and planning systems are quite different systems, but they do share the feature of being a complex system to which a user will ascribe actor behaviour. Therefore, the expectation arises that the results also apply to complex systems in general. Principles which are found here, such as a difference in the ease with which trust is given to a complex system for the different (sub)tasks, clearly have a generic character. For both systems, it appeared that the (sub)tasks which comprised more risk and were more complicated were trusted less readily than (sub)tasks that were simpler and held less risk.

Results such as the fact that a person who interacts with a complex system builds up a certain amount of trust during the course of the interaction can also be called generic results for the interaction with complex systems. One precondition is that the system does not produce remarkable errors.

Furthermore it can be stated that trust, as defined in this research, can be used in other situations in which there is interaction between a person and a complex system. Furthermore, the function of trust that was given in this research also applies to situations in which there is interaction between a person and a complex system.

Finally it can be stated that the way of measuring trust (the four scales of trust) can be applied in a multiple of situations.

7.3 Applicability of the results

The results of this research can be used in various areas. The first area is that which concerns the development of complex systems. During the development of a program, it is important that factors that support trust during working with the program are taken into account. It has been shown that trust in a program aids and speeds up the use of the program. The development of the interaction component has to be looked at, in which the principles, such as keeping the execution of tasks recognisable and supporting as many of the users' goals as possible, are very important. Furthermore, it is very important that the user can make use of a help-facility or a Help-desk, so that the build up of Type I trust is facilitated. In addition, it is of importance that the future users receive a good training in working with a program. Thus, these people can see the program working and can build up Type II trust.

In the concept of having a program support various tasks, the trust of the future user has to be taken into account during the program's development.

It appeared that the greater the feeling of risk on the user's part, the better the trust has to be supported. Therefore it becomes of essential importance that trust in these (sub)tasks is supported by the design of the interaction component.

Thus, trust is of essential importance during the interaction with a complex system. It is possible that the acceptation of a system depends on, among other things, the trust that a system can generate within the future user.

7.4 Background versus Goal

In this section, the difference that was found between the background of the users (test with ZKR) and the users' goals (test with CASH) is discussed. During the tests with the CASH program, the second hypothesis was rejected on the basis of the test results. During the whole test with CASH, in fact, little difference between the commercial group and the non-commercial group was found; the commercial group even showed the reverse of the expectation of the hypothesis on the questionnaire. The commercial group answered more positively on the trust questionnaire. It is possible that a positive 'professional attitude' was expected from the commercial group. This is in contrast to the non-commercial group where a critical attitude was expected, and thus possibly also a more critical attitude regarding the introduced program. It is also very feasible that the commercial group knew the prices of the hearing aids well and, thus, with these, could complete the result in their heads. Thus, the shortcoming of the program in terms of meeting the user's aim was quickly compensated by the users themselves. In this way, the program did meet the users' goals, so no difference could be found between the two groups.

Thus, it is possible that the criterion of meeting or not meeting the user's aim was not clear enough; in other words, the difference between meeting and not meeting the goals was too subtle. This differs in the groups where the difference was the backgrounds. The difference in background had a much clearer relation to the program to be used, and cannot be compensated as easily as the difference in meeting the goals. A person is inextricably linked to his/her background and it cannot be adapted just like that. In the course of working with the program, it appeared that the non-technical group best illustrated the influence of the background. This could be seen in the fact that the non-technical group showed a larger increase in trust than the technical group in the course of working with the program. Therefore it can be expected that when the new users are well supported by a good Help-desk, they are capable of quickly developing a reasonable trust in the new program.

7.5 The future office

If the trend towards more and more complex systems taking over more and more cognitive tasks continues, aspects like trust will assume an increasingly important place. In the future, the applications, which are developed, will possibly have to pay even more attention to how users can be supported in building up trust. In the future office, data warehouses will occupy an increasingly important place (see Gill et al., 1996). These data warehouses will, to an increasing extent, make use of statistical techniques combined with knowledge-based system and neutral networks. It will be clear that many of the management's decisions will be based on the data that are obtained with the aid of the data warehouses. Not only do the data have to be correct and well processed, as this research demonstrates, but the user also has to be able to trust that the results of the system are correct and reliable. During the design phase of such systems, attention will have to be paid to aspects of trust.

Not only systems will change, but also the way of working could also change considerably. Already, a trend towards teleworking can be seen within organisations. This means that the employees will be let loose on applications which support teleworking, such as Internet and Intranet facilities. It is possible that important information will have to be sent to colleagues. The user then has to be able to trust that the information, whether protected or not, will be transmitted well. Packages like e-mail, Internet and intranet systems then will also have to concentrate on supporting the user's trust in the systems which are to be used.

It will be clear that trust will become a more and more important subject during the development of the user dialogue.

7.6 Organisational implications

In order to be able to support trust in a system, organisational changes can be expected in the development of new applications at various places. First, the software house will have to allow scope for the team of designers to pay attention to the actor characteristics of the systems during the design of the interaction dialogue. Particularly the support of trust that will have to be given to the user of the new product will have to be taken into consideration. In addition, the build-up of trust has to be taken into account during the process of compiling the users' documentation. The team that writes the documentation will also have to be accorded the leeway to pay attention to the build-up of the users' trust in the program. This also counts for the development of material for training.

During the introduction of the new system, the future users will have to be allowed the space to build up enough trust during the training session so they can work confidently with the system later on. The organisation into which the system is introduced will have to ensure a good introduction in which attention is paid to the build-up of trust. As mentioned in the section dealing with the build-up of trust, the help of people is especially important in the initial stages. Asking for help is an effective way of supporting a novice user in the process of building up trust. This means that both the organisation that develops the system and the organisation that receives the system will have to make people available who can support the users well, certainly in the initial stages.

7.7 Future research

Of course, in this research, various other interesting subjects have not been dealt with. One of these subjects is an elaborate research on the risk that influences the trust. It has been proved that the perceived risk has an influence on the ease with which trust is given to a complex system. With trust, the extent of the risk estimated by the person is very important. The higher the perceived risk, the more trust that is required. This does not mean that more trust is then given. It does mean that more trust is required to still be able to work well with the actor which is to be trusted. Koller (1988) links trust and risk in a very simple way, but this research suggests that the relation is much more complex. More research on this topic could turn out to be interesting.

Another subject that should be investigated more deeply is the relation between positive and negative events during working with a complex system. In this research, negative events did not cause remarkable fluctuations in trust. Both programs probably worked so well that the users could distinguish neither remarkably negative nor remarkably positive events. In general, it was assumed that a negative event decreases trust and that a positive event increases trust. Theoretically, this seems a tenable premise; however, this relation has not been examined in a way other than with very subjective questionnaires (Muir, 1987; Lee & Moray, 1992).

An other interesting result of this research is that computer systems, from the point of view of the user, tend to have some sort of behaviour and to become part of a social environment. The computer systems are becoming more like players and less like slaves in de social environment of the user. This means that, although no one could say that computers are real living beings, people will still think of them as actors with whom they communicate and develop ‘relationships’. The conclusion is that, as long as people see computers as living beings, the interaction processes should be researched as if they were alive. Following this line of thought, the interaction processes between humans and computers should be researched on a more social point of view, as if they are actors in a social environment.

References

Annett, J., Duncan, K.D., Stammers, R.B., and Gray, M.J., 1971, *Task Analysis,* Training Information Number 6, HMSO, London.

Ballim, A. and Wilks, Y., 1991, Beliefs, Stereotypes and Dynamic Modeling. In: *User Modeling and User-Adapted Interaction.* **1**, Kluwer Academic Publishers, North-Holland, the Netherlands. pp. 33-65.

Bannon, L.M. and Bødker, S., 1991, Beyond the Interface: Encountering Artifacts in Use. In: J.M. Carroll (ed.), *Designing Interaction: Psychology at the Human-Computer Interface.* Cambridge University Press, New York. pp. 227-254.

Barber, B., 1983, *The logic and limits of trust.* Rutgers University Press, New Brunswick, NJ.

Bødker, S., 1989, A Human Activity Approach to User Interfaces. *Human-Computer Interaction*. **4** (3), pp. 171-195.

Bødker, S., 1991, *Through the Interface: A Human Activity Approach to User Interface Design*. Lawrence Earlbaum Associates, Inc., New Jersey.

Booth, P., 1989, *An introduction to Human-Computer Interaction.* Lawrence Erlbaum Associates Inc., New Jersey.

Booth, P. and Brown, G.M., 1989, The Organizational Impact of Computer Systems. In: P. Booth, *An introduction to Human-Computer Interaction.* Lawrence Erlbaum Associates Inc., New Jersey.

Boss, R.W., 1978, Trust and Managerial Problem Solving Revisited. *Group and Organization Studies,* **3**, pp. 331-342.

Breuker, J. A. and Wielinga, B., 1986, Use of models in the interpretation of verbal data. In: *Knowledge Acquisition for Expert Systems: A practical handbook.* Kidd, A. (ed.). Plenum Press, New York. pp. 17 - 44.

Butler, J.K. and Cantrell, R.S., 1984, A Behavioral Decision Theory Approach to Modelling Dyadic Trust in Superiors and Subordinates, *Psychological Reports.* ***55***, pp.19-28.

Card, S.K., Moran, T.P., and Newell, A., 1983, *The psychology of Human-Computer Interaction.* Lawrence Earlbaum Associates Inc., Hillsdale, NJ.

Carroll, J.M., 1991, *Designing Interaction: Psychology at the Human-Computer Interface.* Cambridge University Press. New York.

Chignell, M.H., 1993, Cooperative Human-Machine Reasoning: Communication Through the Interface. In: R.J. Jorna, B. van Heusden & R. Posner, *Signs,Search and Communication: Semiotic Aspects of Artificial Intelligence.* Walter de Gruyter, Berlin.

Chignell, M. and Parsaye, K., 1988, *Expert systems for Experts.* John Wiley and Sons, Inc.

Cleal, D.M. and Heaton, N.O., 1988, *Knowledge-based systems.* Ellis Horwood Limited.

Dahlbom, B. and Mathiassen, L., 1993, *Computers in Context: The Philosophy and Practice of Systems Design.* Blackwell Publishers, Cambridge/USA.

Diaper, D. and Johnson, P., 1989, Task Analysis for Knowledge Descriptions: Theory and application in Training. In: J. Long and A. Whitefield (eds.), *Cognitive Ergonomics and Human-Computer Interaction.* Cambridge University Press, Cambridge.

Dzida, W., 1987, On tools and interfaces. In: M. Frese, E. Ulich and W. Dzida (eds.) *Psychological Issues of Human-Computer Interaction in the Work Place.* Elsevier Science Publishers B.V., North-Holland, the Netherlands. pp. 339-355.

Early, P.C., 1986, Trust, Perceived Importance of Praise, Criticism and Work Performance: An Examination of feedback in the United States and England. In: *Journal of Management*, **12**, pp. 457-473.

Eberts, R.E., 1994, *User Interface Design*, Prentice-Hall, Inc., Englewood Cliffs, New Jersey.

Ericsson, K. A. and Simon, H. A., 1984, *Protocol Analysis: Verbal Reports as Data.* The MIT Press, Cambridge, MA.

Fisher, S., 1985, Control and Blue Collar Work. In: C.L. Cooper and M.J. Smith (eds.) *Job Stress and Blue Collar Work.* John Wiley & Sons Ltd.

Fox, J., 1987, Making decisions under the influence of knowledge. In: P.Morris: Modelling cognition. John Wiley & Sons Ltd., New York. pp. 199-212

Frese, M., 1987, A theory of control and complexity: Implications for software design and integration of computer systems into the workplace. In: M. Frese, E. Ulich & W. Dzida, *Psychological issues of Human Computer Interaction in the Work Place.* Elsevier Science Publishers B.V.-North-Holland, the Netherlands. pp. 313-337

Frese, M. and Brodbeck, F., 1989, *Computer in Buro und Verwaltung. (Computers in the office and Administration).* Springer Verlag, Berlin.

Frese, M. and Zapf, D., 1993, Action as a Core of Work Psychology: A German Approach. In: D.M. Dunette, L.M. Hough and H.C. Triandis (eds.), *Handbook of industrial and Organizational Psychology,* **4**, Consulting Psychologists Press, Paolo Alto.

Gardiner, M.M. and Christie, B., 1987, *Applying Cognitive Psychology to User-Interface Design.* John Wiley & Sons Ltd., Chichester.

Giddens, A., 1990, *The Consequences of Modernity*. Stanford University Press, Stanford CA.

Gill. H.S. and Rao, P.C., 1996, *The official Client/Server Computing Guide to Data Warehousing.* Que Corporation, Indianapolis.

Green, R., Day, S., Banford, J., 1989-a, A comparative evaluation of four hearing-aid selection procedures. I-Speech discrimination measures of benefit. In: *British Journal of Audiology*. **23**.

Green, R., Day, S., Banford, J., 1989-b, A comparative evaluation of four hearing-aid selection procedures. II-Quality judgements as measures of benefit. In: *British Journal of Audiology.* **23**.

Groen, J.J., 1971, Slechthorendheid en hoortoestellen, Stafleu's wetenschappelijke uitgeversmij. N.V., Leiden, the Netherlands.

Hart, K., 1988, Kinship, Contract and Trust: The Economic Organization of Migrants in an African City Slum. In: D. Gambetta, *Trust: Making and Breaking Co-operative Relations.* Blackwell Ltd., Oxford.

Hart, M., 1989, *Knowledge Acquisition for Expert systems*. Kogan Page, London.

Heimovics, R.D., 1984, Trust and the Influence in an Ambiguous Group setting. In: *Small Group Behavior,* **15**, pp. 545-552.

Hollon, C.J. and Gemmil, G.R., 1977, Interpersonal Trust and Personal Effectiveness in the Work Environment. In: *Psychological Reports*, **40**.

Howard, N., 1994, *Drama Theory and HCI.* In: e-Mail message from e-Mail list of EACE (European Association of Cognitive Ergonomics) (eace-bcs@cs.vu.nl).

Hughes, J., King, V., Randall, D., and Sharrock, W., 1993, Ethnography for System Design: A Guide. Working Paper, COMIC-LANCS-2-4.

Johnson, P., 1992 *Human-Computer Interaction: Psychology, Task analysis and Software Engineering.* McGraw-Hill Book Company Europe, London.

Jorna, R.J., Gazendam, H.W.M., Heesen, H.C. and Wezel W.M.C. van, 1996, *Plannen en roosteren. Taakgericht analyseren, ontwerpen en ondersteunen.* Leiderdorp: Lansa Publishing B.V., the Netherlands.

Kapteyn, T.S., Clemens, A., Glazenburg, B.E., *Slechthorende en hoortoestel*. Nederlandse vereninging voor audiologie. Loenen; Utrecht, the Netherlands.

Kidd, A.L., 1986, Knowledge acquisition: An introductory framework. In: *Knowledge Acquisition for Expert Systems: A practical handbook.* Kidd, A. L. (ed.). Plenum Press, New York. pp. 1-16.

Kieras, D. and Polson, P.G., 1986, An approach to the formal analysis of user complexity. In: *International Journal of Man-Machine Studies*, **22**, pp. 365-394.

Koller, M., 1988, Risk as a determinant of Trust. In: *Basic applied social psychology.* **9**(4), pp. 265-276.

Kuutti, K. and Karasti, H., 1994, Bridging the “Great Divide” in CSCW Research and Development: Ethnography, Participatory Design and Developmental Work Research. In: *Proceedings of Schärding workshop: Interdisciplinary design of computer systems.* Schärding, Austria.

Lee, J. and Moray, N., 1992, Trust, Control Strategies and Allocation of Function in Human-Machine Systems. In: *Ergonomics*, **35**, pp. 1243-1270.

Lewis, C., 1991, Inner and Outer Theory in Human-Computer Interaction. In: J.M. Carroll (ed.), *Designing Interaction: Psychology at the Human-Computer Interface.* Cambridge University Press, New York. pp. 154-162.

Lucas, P. J. F. and Gaag, L. C., van der, 1988, *Principes van Expert-systemen*. Academic Service, Schoonhoven, the Netherlands.

Lueke, E., Pagerey, P.D., Brown, C.R., 1987, User requirements-gathering through verbal protocol analysis. In: *Cognitive engineering in the design of human-computer interaction and expert systems*. Salvendy, G. (ed.). **Vol 2**, Elsevier Science Publishers B.V., Amsterdam. pp. 523 -528.

Luhmann, N., 1979, *Trust and Power: Two Works by Niklas Luhmann.* John Wiley & Sons, Chichester.

Luhmann, N., 1988, Familiarity, Confidence, Trust: Problems and Alternatives. In: D. Gambetta, *Trust: Making and Breaking Cooperative Relations.* Blackwell Ltd., Oxford. pp. 94-107

Maurer, M.M., 1983, *Development and validation of a measure of computer anxiety.* Unpublished master thesis, Iowa Sate University, Ames, IA.

Maurer, M.M., 1984, *Computer Anxiety Correlates and what they tell us: A literature review.* In: Computers in Human Behavior, **10** (3), pp. 369-376

Maurer, M.M., 1994, The reduction of Computer Anxiety: Its relation to relaxation training, previous computer coursework, achievement and need for cognition. In: *Journal of research on computing in education*, **26** (2), pp. 205-219

McGinnies, E. and Ward C.D., 1980, Better Liked than Right: Trustworthiness and Expertise as Factors of Credibility. In: *Personality and Social Psychology Bulletin*, **6**, pp. 467-472.

McGraw, K.L., 1992, *Designing and evaluating user interfaces for knowledge-based systems.* Ellis Horwood, New York.

Meyers, B.A.,1988, *Creating User Interfaces by Demonstration.* Academic Press, Inc., London.

Mietus, D.M., 1994, *Understanding planning for effective decision support: A cognitive task analysis of nurse scheduling.* Doctoral dissertation, Faculty of Management and Organization, University of Groningen, the Netherlands.

Moran, T.P., 1981, The command language grammar: a representation for the user interface of interactive computer systems. In: *International Journal of Man-Machine Studies,* **15**, pp. 3-50.

Muir, B.M., 1987, Trust between humans and machines, and the design of decision aids. In: *International Journal of Man-Machine Studies*, **27**, pp. 527-539.

Mulder, G., Lamain, W., and Passchier, P., (1992), De cognitieve interface. In: R.J. Jorna and J.L. Simons, *Kennis in Organisaties: Toepassingen en Theorie van Kennissystemen*. Couthino BV, Muiderberg, the Netherlands.

Nickell, G.S. and Pinto, J.N., 1986, The computer attitude scale. In: *Computers in human behavior,* **2**, pp. 301-306.

Nickerson, R.S., 1986, *Using Computers: Human factors in Information systems.* MIT Press, Cambridge, Massachusetts.

Norman D.A., 1986**,** Cognitive engineering**.** In: D.A. Norman & S.W. Draper, *User-centred system design: New perspectives on human-computer interaction*. Erlbaum. Hillsdale, New Jersey.

Numan, J.H., 1991, *C.A.S.H.* Intern report Instituut voor Experimentele Psychologie (Institute for Experimental Psychology), Groningen, the Netherlands (In Dutch).

Numan, J.H., 1992a, *Functionele specificatie backtrack mechanisme ZKR.* Groningen, the Netherlands (In Dutch).

Numan, J.H., 1992b, *Technische specificatie backtrack mechanisme ZKR.* Groningen, the Netherlands (In Dutch).

Numan, J.H., 1994a, *Gebruikershandleiding ZKR.* Groningen, the Netherlands (In Dutch).

Numan, J.H., 1994b, *Ontwikkeling van een kennissysteem voor de aanmeting van hoortoestellen.(Development of a knowledge-based system for selection of hearing-aids),* Intern report Instituut voor KennisSystemen (Institute for Knowledgebased Systems) (IKS), Groningen, the Netherlands (In Dutch).

Numan, J. H., 1994c, Computer ondersteunde selectie van hoortoestellen voor patiënten met een hoorstoornis. In: W.P.A. Beckers & A.J. ten Hoopen, *Ontwikkelingen in de Medische Informatica; Proceedings van het Medische Informatica Congres*, 25 and 26 November 1994, Veldhoven. Krips Repro, Meppel, the Netherlands. ISBN 90-802230-1-8. (In Dutch).

Numan, J. H. and Oldenkamp, J.H., 1994d, Dienstroosterplanning voor kinderartsen door een kennissysteem. In: W.P.A. Beckers & A.J. ten Hoopen, *Ontwikkelingen in de Medische Informatica; Proceedings van het Medische Informatica Congres,* 25 and 26 November 1994, Veldhoven. Krips Repro, Meppel, the Netherlands. ISBN 90-802230-1-8. (In Dutch).

Numan, J. H., Jorna, R., and Arion, M., 1994e, Designing Knowledge-based Systems with the Aid of HCI. In: *Proceedings of Schärding workshop: Interdisciplinary design of computer systems.* Schärding, Austria.

Oldenkamp, J.H., 1993, *Definities van kennissystemen.* Research Report number: RR 1993-12. Faculty of Management and Organization, University of Groningen, the Netherlands.

O'Reilly C.A., 1978, The Intentional Distortion of Information in Organizational Communication: A Laboratory and Field Investigation. In: *Human Relations*, **31**, pp. 173-193.

O'Reilly C.A. and Roberts K.H., 1974, Information Filtration in Organizations: Three Experiments. In: *Organizational Behavior and Human Performance,* **11**, pp. 253-265.

O'Reilly C.A. and Roberts K.H., 1976, Relationships Among Components of Credibility and Communication Behaviors in Work Units. In: *Journal of Applied Psychology.* **61**, pp. 99-102.

Payne, S.J. and Green, T.R.G., 1986, Task Action Grammars. In: *Human Computer Interaction.* **2**, pp. 93-133.

Petermann, F., 1992, *Psychologie des Vertrauens-2.,* Quintessenz-Verl.-GmbH, Munich.

Poggi, G., 1979, Introduction. In: N. Luhmann, *Trust and power: Two works by Niklas Luhmann.* John Wiley & Sons, Chichester.

Pycock, J., Calvey, D., Sharrock, W., King, V., and Hughes, J., 1994, Present in the Plan: Process Models and Ethnography. Working Paper, COMIC-MAN-1-7.

Reitman-Olson, J. and Reuter, H.H., 1987, Extracting expertise from experts: Methods for knowledge acquisition. In: *Expert Systems*. **4** (3). pp. 152 - 168.

Rempel, J.K., Holmes, J. G., and Zanna, M.P., 1985, Trust in Close Relationships. In: *Journal of Personality and Social Psychology*, **49**, pp. 95-112.

Roe, R.A., 1988, Acting system design - an Alternative Approach to the Design of Man-Computer Systems. In: V. de Keyser, T. Qvale, B. Wilpert & S.A. Ruiz Quintanilla, *The Meaning of Work Technological Options,* John Wiley & Sons. New York, pp. 179-195.

Searle, J.R., 1980, Minds, brains and programs. In: *The Behavioral and Brain Sciences, Vol. 3* . Cambridge University Press. pp. 417-457.

Shneiderman, B., 1992, *Designing the User Interface: Strategies for Effective Human-Computer Interaction.* Addison-Wesley Publishing Company, Inc., Massachusetts.

Thimbleby, H., 1990, *User Interface Design.* Addison-Wesley, Massachusetts.

Versendaal, J., 1991, *Separation of the User Interface and Application.* Thesis, Delft University, the Netherlands.

Wærn, Y., 1989, *Cognitive aspects of Computer-Supported tasks.* John Wiley & Sons Ltd., Chichester.

Weir, G.R.S., 1991, Living with Complex Interactive Systems. In: G.R.S. Weir & J.L. Alty, *Human-Computer Interaction and Complex Systems.* Academic Press Ltd., London. pp. 1-22.

Wielinga, B.J. and Breuker, J.A., 1988, Interpretation of verbal data for knowledge acquisition. In: *Human-Computer Interaction: psychonomic aspects*. G.C. van der Veer & G. Mulder (eds.). Springer-Verlag Berlin, Heidelberg, Germany.

Wit, H.P., 1985 *Postacademische cursus audiologie*. Rijksuniversiteit Groningen, Audiologisch Instituut, Groningen, the Netherlands.

Woods, D.D., 1988, Coping with Complexity: The Psychology of Human Behaviour in Complex Systems. In: L.P. Goodstein, H.B. Anderson & S.E. Olsen: *Tasks, Errors and Mental Models.* Taylor and Francis, London. pp. 128-148.

Zand, D.E., 1972, Trust and Managerial Problem Solving. *Administrative Science Quarterly,* **17** pp. 229-239.

Zuboff, S., 1988, *In the age of the Smart Machine: The future of work and power.* Basic Books, New York.

Samenvatting

Dit onderzoek gaat over vertrouwen van gebruikers in complexe software, met name kennissystemen. Deze systemen zijn zo complex geworden dat het voor een gebruiker niet meer inzichtelijk is wat het programma doet. Ondanks deze gecompliceerdheid wordt er wel met deze programma's gewerkt. Met andere woorden de gebruikers vertrouwen er op dat de programma's doen wat ze horen te doen. De complexiteit is mede een gevolg van het feit dat bepaalde cognitieve taken van een gebruiker zijn overgenomen. Het programma begint in de ogen van de gebruiker bepaalde actor kenmerken te vertonen (een belangrijk kenmerk van een actor is dat deze een zekere mate van autonomie bezit en daarmee een zekere mate van handelingsvrijheid). Deze actor kenmerken maken dat de computer 'menselijker' op de gebruiker overkomt en dat tevens het gevoel ontstaat dat het programma een zekere handelingsvrijheid geniet. Kernwoord in dit onderzoek is het concept vertrouwen. In het werken met complexe systemen hanteren gebruikers dit vertrouwen onder andere door op een verstandige wijze te reageren op het gedrag van deze systemen. Dit komt allereerst tot uiting in de mens-machine communicatie en dan vooral in de Human-Computer Interactie. De hoofdvraag van dit onderzoek is daarmee:

> *Wanneer vertrouwen een belangrijk concept is bij mens-computer interactie, wat is dan dit vertrouwen, kan een model van vertrouwen worden opgesteld en welke maten kunnen worden geïdentificeerd om vertrouwen te kunnen meten?*

Deze samengestelde vraag bestaat uit de volgende deelvragen:

1a: Hoe kan vertrouwen worden omschreven en gebruikt?
1b: Welke functie heeft vertrouwen?
1c: Kan een model van vertrouwen worden opgesteld?
1d: Welke maten kunnen worden geïdentificeerd om vertrouwen te kunnen meten?
2a: Hoe wordt vertrouwen in een programma opgebouwd?
2b: Hoe kunnen de dynamische aspecten worden gemeten?
3: Welke aspecten hebben invloed op het vertrouwen van een gebruiker in een systeem?

In hoofdstuk 1 wordt ingegaan op de achtergrond van de onderzoeksvragen. Het blijkt dat vertrouwen een belangrijk onderwerp is bij mens-computer interactie, maar dat weinig onderzoek is gedaan naar het onderwerp vertrouwen tussen mens en computer programma. In hoofdstuk 2 wordt daarom ook dieper ingegaan op het mens-computer interactie onderzoek Er worden vier invalshoeken besproken, te weten:

- de taakgerichte invalshoek,
- de machinegerichte invalshoek,
- de gebruikersgerichte invalshoek,
- de (sociale) contextgerichte invalshoek.

Deze vier invalshoeken zijn alle vier in meer of mindere mate van belang bij het ontwerpen van een interactiecomponent, tot uitdrukking gebracht in de gebruikersinterface.

In hoofdstuk 2 wordt duidelijk gemaakt dat ondanks de grote waarde van deze invalshoeken er toch nog een aspect is blijven liggen waar weinig aandacht aan werd geschonken, namelijk vertrouwen. Om vertrouwen te kunnen onderzoeken moet ten eerste antwoord worden gegeven op de vraag: wat is vertrouwen en ten tweede op de vraag: welke functie heeft vertrouwen.

Om antwoord te kunnen geven op deze vragen wordt in hoofdstuk 3 ingegaan op onderzoek naar vertrouwen en wordt een antwoord gegeven op de definitie van vertrouwen in sociaal-psychologisch onderzoek. Daarna is een model en een definitie van vertrouwen gegeven zoals dat in dit onderzoek is gebruikt. Daarnaast wordt gekeken naar aan vertrouwen gerelateerde concepten (zoals attitudes, computerangst, complexiteit en geloof) en wordt het begrip vertrouwen ingekaderd (wat is het wel en wat niet) Verder wordt aangegeven wat de invloed is van vertrouwen op de genoemde begrippen. Tenslotte worden in hoofdstuk drie de twee belangrijkste hypothesen opgesteld. Deze hypothesen zijn:

Hypothese 1

> Wanneer een kennissysteem wordt geïntroduceerd dan zal wanneer er sprake is van de bekendheid met (kennis van) technisch hoogwaardige apparaten het vertrouwen in het kennissysteem hoger zijn dan wanneer deze bekendheid (kennis) niet aanwezig is.

Hypothese 2

> Wanneer een kennissysteem niet alle belangrijke doelen van een gebruiker zal ondersteunen dan zal het vertrouwen in het kennissysteem lager zijn dan wanneer alle doelen van een gebruiker wel worden ondersteund.

Hypothese 1 heeft als belangrijkste uitgangspunt dat vertrouwen een basis van bekendheid met de te vertrouwen actor nodig heeft. Er wordt dan ook gebruik gemaakt bij de testen van een technische en niet-technische groep. Hypothese 2 heeft als belangrijkste uitgangspunt dat de doelen van een vertrouwende persoon moeten worden ondersteund door de te vertrouwen actor. Er wordt gebruik gemaakt van testen met een commerciële groep en een niet-commerciële groep proefpersonen.

In hoofdstuk 4 wordt ingegaan op de metingen en het onderzoeksontwerp. In dit hoofdstuk zijn de vier maten geïdentificeerd die het concept vertrouwen moeten meten. De maten zijn:

1. Basis vertrouwen (attitudes)	(TRQUEST)
2. Vertrouwen fluctuaties tijdens de interactie	(TRCATEGO)
3. Vertrouwende handelingen	(TRBEHAV)
4. Gevolgen van vertrouwen	(TRREACT)

In hoofdstuk 5 en 6 worden de twee hypothesen uit hoofdstuk 3 uitgewerkt met behulp van testen met proefpersonen die werkten met twee verschillende kennissystemen. In hoofdstuk 5 werd hypothese 1 uitgewerkt terwijl in hoofdstuk 6 de tweede hypothese werd uitgewerkt.

Voor de test die is beschreven in hoofdstuk 5 is gebruik gemaakt van het kennissysteem ZKR (ZieKenhuis Roostering) wat een planningssysteem is. De proefpersonen bestaan uit twee verschillende groepen. Groep 1 is een niet-technische groep terwijl groep 2 een technische groep van proefpersonen is. In dit hoofdstuk blijkt dat de hypothese 1 kan worden ondersteund.

Voor de test die is beschreven in hoofdstuk 6 is gebruik gemaakt van het kennissysteem CASH (Computer Aided Selection of Hearing-aids) wat een diagnose of selectiesysteem is. De proefpersonen voor het testen van hypothese twee bestonden ook uit twee groepen. Groep 1 waren proefpersonen werkzaam in de commerciële branche, terwijl de tweede groep bestond uit proefpersonen uit de niet-commerciële branche. In dit hoofdstuk blijkt dat de hypothese 2 niet kan worden ondersteund.

Hoofdstuk 7 tenslotte geeft met betrekking tot de gevonden resultaten, uit de hoofdstukken 5 en 6, verschillende conclusies. Belangrijke conclusies zijn dat vertrouwen veranderd in de tijd en dat vertrouwen verschilt tussen verschillende typen taken.

Vertrouwen blijkt een zeer sterk mechanisme te zijn waar mensen gebruik van maken wanneer ze zich bevinden in gecompliceerde, onzekere en/of risicovolle situaties.

Verder blijkt dat door het veranderende karakter van de huidige computerprogramma's er behoefte ontstaat aan uitbreiding van het onderzoek naar mens-computer interactie. Onderzoek naar vertrouwen tussen mens en machine is een uitstekende manier om inzicht te krijgen in het interactie proces tussen mens en (complexe) machine, zeker waar die machine zodanige kenmerken gaat vertonen dat ze 'menselijk' lijkt te worden.

Stellingen behorende bij het proefschrift

Knowledge-Based Systems as Companions:
Trust, Human Computer Interaction and Complex Systems

van

John Han Numan

1. Door de invoering van steeds complexere en abstractere systemen kunnen en moeten mensen meer effectieve strategieën gebruiken om deze complexiteit te reduceren. Vertrouwen is een manier om deze reductie te bewerkstelligen.
2. Uiterlijk vertoon kan zeer ten onrechte vertrouwen bij het publiek opwekken, getuige de mediapresentatie van verschillende politici. Software vormt in deze geen uitzondering.
3. Wanneer gebruikers van een programma veel fouten maken verdient het aanbeveling het programma aan te passen en niet de gebruiker.
4. De overgang van handmatig roosteren naar computerondersteund roosteren is vergelijkbaar met het overgaan van handmatig schrijven naar tekstverwerking. Het grote verschil is dat de weerstand van de overgang naar computerondersteund roosteren veel groter is.
5. Ondersteuning van het vertrouwen van een gebruiker in een computersysteem zal de snelheid van het gebruik van dit computersysteem doen toenemen.
6. Computersystemen worden meer en meer gecompliceerd. Dit heeft tot gevolg dat gebruikers steeds minder weten van de werking van deze systemen. Hierdoor moeten gebruikers van deze systemen er steeds meer op vertrouwen dat het systeem zal doen wat het moet doen.
7. Een leeg bureau is dodelijk voor serendipiteit.
8. Door de vergroting van de mogelijkheden van computersystemen krijgen deze systemen in de ogen van gebruikers een grotere handelingsvrijheid en een steeds menselijker karakter. Er zal steeds meer behoefte ontstaan aan onderzoek naar ‘menselijke’ interactie elementen in het vakgebied van Mens-Computer Interactie.
9. Bij de acquisitie van kennis voor het ontwikkelen van een diagnose systeem, zoals CASH, is het van essentieel belang dat de experts niet de geringste indruk krijgen dat het systeem hen zal vervangen.
10. Uitbreidingen van de hedendaagse (computer) technologie geeft mensen minder directe controle op de omgeving. Het vertrouwen in deze technologie zal dus steeds groter moeten worden.
11. Het aantal comfortabele stoelen in de wachtruimte van het gemeentehuis Groningen doet het ergste vermoeden van de wachttijden.